applied sciences

MDPI

Article

Automatic Detection of Clickbait Headlines Using Semantic Analysis and Machine Learning Techniques

Mark Bronakowski [1], Mahmood Al-khassaweneh [1,*] and Ali Al Bataineh [2]

[1] Engineering, Computing and Mathematical Sciences, Lewis University, Romeoville, IL 60446, USA
[2] Department of Electrical and Computer Engineering, Norwich University, Northfield, VT 05663, USA
* Correspondence: malkhassaweneh@lewisu.edu

Abstract: Clickbait headlines are misleading headiness designed to attract attention and entice users to click on the link. Links can host malware, trojans and phishing attacks. Clickbaiting is one of the more subtle methods used by hackers and scammers. For these reasons, clickbait is a serious issue that must be addressed. This paper presents a method for identifying clickbait headlines using semantic analysis and machine learning techniques. The method involves analyzing thirty unique semantic features and exploring six different machine learning classification algorithms individually and in ensemble forms. Results show that the top models have an accuracy of 98% in classifying clickbait headlines. The proposed models can serve as a template for developing practical applications to detect clickbait headlines automatically.

Keywords: clickbait; classification; machine learning; semantic analysis

Citation: Bronakowski, M.; Al-khassaweneh, M.; Al Bataineh, A. Automatic Detection of Clickbait Headlines Using Semantic Analysis and Machine Learning Techniques. *Appl. Sci.* **2023**, *13*, 2456. https://doi.org/10.3390/app13042456

Academic Editor: Luis Javier Garcia Villalba

Received: 1 February 2023
Revised: 7 February 2023
Accepted: 10 February 2023
Published: 14 February 2023

1. Introduction

As a result of the proliferation of information online, we are now subjected to a barrage of advertisements and news headlines on virtually every page we access. Since so many people have access to the internet, websites and news outlets are constantly competing for viewers. As a result, they are under pressure to create ever more appealing, catchy and provocative article headlines, regardless of their accuracy. Because of this, there has been a rise in recent years of sensationalist "news" headlines that do not tell readers anything valuable about the story but are meant to grab their attention [1]. The term 'clickbait' refers to misleading links with sensationalized headlines that intend to attract the viewers' attention and entice them to click on the link [2]. In a broad sense, clickbait headlines meet two main criteria: (1) they mislead readers about the article's contents, and (2) they take advantage of the so-called "curiosity gap" by not explaining the entire article's contents. To put it more simply, the text included in these headlines either makes the reader curious about the rest of the article's contents or discusses topics that are not addressed in the body of the article itself. It is imperative that we make a clear distinction between clickbait and fake news, a topic that has been receiving a growing amount of attention as of late. The difference lies in the fact that fake news purposefully presents its audience with information that they should know is false in order to gain their trust. On the other hand, clickbait almost always just contains "junk" news that lacks any real journalistic integrity and is not designed to trick the reader into believing false claims. Clickbait can pose a threat to Internet users and has become more prevalent across the web, not just on less reputable sites [3]. Recent research from Stanford University highlights how clickbait is making its way onto more reputable journalism sites [4]. Clickbait can have an even more malicious purpose such as phishing for personal information, or even worse, hosting malware. For these reasons, clickbait is a serious issue that must be addressed. The first step in addressing this problem is to distinguish clickbait headlines from true headline links. On the surface, clickbait headlines can be hard to recognize, as they are

designed to fool the user, but there are key semantic features that can help identify clickbait headlines. AI tools based on machine learning algorithms can detect and block clickbait in a systematic manner. Machine learning unlocks the power of data in novel ways [5]. This technology assists computer systems in learning from and improving on their experiences by creating computer programs that can automatically access data and perform tasks through predictions and detections. As you feed more data into a machine, the algorithms learn more about the machine, which improves the results [6]. Several works based on machine learning techniques have been proposed for clickbait classification.

Razaque et al. [7] developed ClickBaitSecurity to distinguish between legitimate and illegitimate links by accurately using a recurrent neural network (RNN). In comparison to existing solutions, the test results showed that their proposed model has high accuracy in detecting malicious and safe links.

Shang et al. [8] introduced a content-agnostic scheme, Online Video Clickbait Protector (OVCP), to effectively detect clickbait videos by analyzing the comments left by viewers of the video. Unlike other solutions, OVCP does not directly analyze the video's content and pre-click data. As a result, it is resistant to sophisticated content creators who frequently create clickbait videos that can evade current clickbait detectors. Their experiments proved that OVCP could accurately identify clickbait videos.

Using social media datasets, Liao et al. [9] proposed federated hierarchical hybrid networks to build clickbait detection models. The titles and contents are stored by different parties whose relationships must be exploited for clickbait detection. In comparison to other cutting-edge methods, their proposed approach demonstrated high efficacy.

Agrawal et al. [10] introduced compiled clickbait corpus and proposed a model for detecting clickbait using convolutional neural networks (CNN). The corpus was built using various social media platforms and deep learning for learning features. The model outperformed other models in detecting clickbait.

Setlur et al. [11] presented a semi-supervised classification-based approach utilizing attentions sampled from a Gumbel–Softmax distribution. An additional loss over the attention weights was applied to encode prior knowledge. The authors also presented a confidence network, which enables learning over weak labels and improves resiliency to noisy labels. According to the results, the model achieved over 97% accuracy with only 30% of strongly labeled samples.

Fakhruzzaman et al. [12] proposed a based neural network classifier with a pre-trained language multilingual bidirectional encoder representations from transformers (M-BERT) model to classify clickbait and non-clickbait headlines. The model was evaluated on a dataset of 6632 headlines using the five-fold cross-validation approach achieving an f1-score of 0.914.

Thomas et al. [13] presented a system based on the fusion of neural networks, which incorporates various forms of available data. The proposed system requires no linguistic preprocessing and generalizes to new domains and languages more quickly. The model achieves an f1 score of 0.564.

Kumar et al. [14] proposed a bidirectional LSTM with an attention mechanism to learn the extent to which a word contributes to the clickbait score of a social media post in a different way. They also used a Siamese net to detect similarities between the source and target data. To add another layer of complexity to the model, they also use CNN to learn image embeddings from large amounts of data. Their experiments were carried out on a test corpus of 19538 social media posts and they achieved an F1 score of 0.65.

Cao et al. [15] used a random forest regression algorithm to create a computational clickbait detection system. A dataset of over 21,000 headlines/titles was used and the 60 most relevant features were extracted. On the clickbait class, the model achieved an f1 score of 0.61.

While previous studies have attempted to address this issue, they have limitations. For example, some studies rely solely on lexical analysis or shallow features, which may not fully capture the semantic meaning of headlines. Other studies do not consider the

impact of different machine learning techniques on classification accuracy. To address these limitations, this paper presents an effective method to categorize clickbait and non-clickbait headlines using semantic analysis and machine learning techniques. Thirty unique semantic features were investigated and six different machine learning classification algorithms were explored individually and as ensembles. The classification algorithms utilized are decision tree, logistic regression, naïve Bayes, support vector machine, k-nearest neighbor and gradient-boosted decision tree. These algorithms were selected because they are widely used in the field of text classification and have been shown to produce good results. The selection of these algorithms was based on the available literature and past studies that have used these algorithms to perform text classification tasks. Additionally, these algorithms represent a diverse range of techniques and approaches, which allows us to evaluate the effectiveness of different methods and to identify the best approach for categorizing clickbait and non-clickbait headlines. To train, test and validate the six algorithms, a large dataset of 32,000 sample headlines collected from different news websites was used. The dataset contained a 50/50 mix of clickbait and non-clickbait headlines.

The main contributions of this paper can be summarized as follows:

1. A method for identifying clickbait headlines using semantic analysis and machine learning techniques is presented.
2. Thirty unique semantic features are investigated and six different machine learning classification algorithms (decision tree, logistic regression, naïve Bayes, support vector machine, k-nearest neighbor and gradient-boosted decision tree) are explored, both individually and as ensembles.
3. A large dataset of 32,000 sample headlines collected from different news websites is used to train, test and validate the techniques; this dataset has a 50/50 mix of clickbait and non-clickbait headlines.

This paper is organized as follows. Section 2 discusses the composition and details of the dataset used. In Section 3, previous research on the topic is examined to identify key semantic features that are commonly associated with clickbait headlines. The correlation between these features and clickbait headlines is analyzed in Section 4. The classification method using different models, both individually and in ensemble form, along with their results, is described in Section 5. In Section 6, the accuracy of the models is compared with similar studies. Finally, the conclusions and findings of the research are presented in Section 7, including a discussion of the limitations of the study and suggestions for future research.

2. Dataset

The headline dataset utilized in this project was downloaded from Kaggle [16], an online community of data scientists and machine learning practitioners. The dataset consists of 32,000 headlines collected from various news sites and contains 15,999 clickbait and 16,001 non-clickbait headlines. The clickbait headlines were collected from sites such as 'BuzzFeed', 'Upworthy', 'ViralNova', 'Thatscoop', 'Scoopwhoop' and 'ViralStories'. The non-clickbait headlines were collected from trustworthy news sites such as 'WikiNews', 'New York Times', 'The Guardian' and 'The Hindu'. The dataset is comprised of two feature columns: (1) the "Headline" feature contains headlines from news sites in text format; (2) the "Clickbait" feature contains binary numeric labels, 1 = clickbait and 0 = non-clickbait. There are no missing data elements for the 32,000 instances.

3. Feature Formulation

Related works in the area of clickbait classification offer insight into semantic features that occur more frequently in clickbait headlines compared to non-clickbait headlines. This section leverages these related works to identify 30 key semantic features linked to clickbait headlines for use in classification modeling. Semantic features associated with clickbait include sentence structure, parts-of-speech, forward referencing, punctuation, common clickbait words and informality. The classification approach used in this project

focuses on analyzing the semantic styles of the text in headlines and not the content of the linked pages.

Chakraborty et al. [1] found that clickbait headlines typically have a greater word count than conventional non-clickbait headlines. In addition, they determined that even though clickbait headlines have more words, the average word length is shorter. They also recognized that stop words, the most common English words, occur more frequently in clickbait headlines. In addition, they concluded clickbait headlines often employ determiners and contractions. Determiners are comprised of articles (*a/an, the*), demonstratives (*this, that, these, those*), possessives (*my, your, his, her, its, our, their*) and quantifiers (*many, much, more, most, some*).

Similarly, Blom et al. [17] postulate that forward referencing is another key semantic headline style used to create anticipation and curiosity to lure readers to click. Forward referencing refers to referencing forthcoming parts of the headline upfront or using a word that gets its meaning from a subsequent word or phrase. Forward referencing can be identified by the presence of demonstrative pronouns (*this, that, these, those*), personal pronouns (*I, you, he, she, it, we, they, me, him, her, us and them*), superlative adverbs (*–est, –ly*) and definite articles (*the*).

Biyani et al. [18] describe how clickbait headlines are made more attention grabbing with the use of acronyms, numbers, upper case letters, questions, quotes, exclamations and other punctuation patterns. They determined that clickbait headlines are more likely to begin with 5W1H words (what, why, when, who, which, how) than non-clickbait headlines. They also observed that the language of clickbait headlines tends to be less formal than that of conventional non-clickbait headlines. To capture this difference in informality Biyani et al. utilized four indices that measure the readability/informality level of text. The Coleman Liau Index [19] for readability is based on the number of letters per word and words per sentence. The $CL\ Index$ is computed by:

$$CL\ Index = 0.0588L - 0.296S - 15.8, \tag{1}$$

where L is the average number of letters per 100 words and S is the average number of sentences per 100 words.

Anderson's RIX Readability Index, $RIX\ Index$, is a simplified version of Bjornsson's LIX Readability Index [20], $LIX\ Index$. Both indices are based on the number of words per sentence. The indices are computed by:

$$LIX\ Index = \frac{W}{S} + \frac{100LW}{W}, \tag{2}$$

and

$$RIX\ Index = \frac{LW}{S}, \tag{3}$$

where W is the number of words, LW is the number of long words (7 or more letters) and S is the number of sentences.

The formality measure index (**F-Score**) developed by Heylighen and Dewaele [21] provides a measure for formality based on the frequencies of different parts of speech of words in the text. They found nouns, adjectives, articles and prepositions are more frequent in formal styles; pronouns, adverbs, verbs and interjections are more frequent in informal styles. The F-Score is computed by:

$$\begin{aligned}\textbf{F-Score} = (&noun\ freq + adjective\ freq \\ &+ preposition\ freq + article\ freq \\ &pronoun\ freq - verb\ freq\ adverb\ freq \\ &interjection\ freq + 100)/2.\end{aligned} \tag{4}$$

Biyani et al. also determined the key words: "reason", "why", "just", "this" and "one" have a high frequency of occurrence in clickbait headlines.

Of the 30 key semantic features formulated above, 22 features are comprised of binary values (1 = presence and 0 = absence of the feature). The four readability/informality and two ratios plus the word count and average word length features are continuous numeric values. A summary of the 30 key semantic features to be used in the classification of clickbait headlines is presented in Table 1.

Table 1. Key semantic classification features.

Count Features	**Forward Referencing Features**	**Readability/Informality**
Word count	Begins w/ determiner/superlative	CL Index
Aver word length	Contains determiner/superlative	LIX index
Ratio of stop-words	Contains demonstrative	RIX index
Ratio words begin w/uppercase	Contains possessive/pronoun	F-Score
Misc. features	**Key Word features**	**Punctuation features**
Begins w/ number	Begins w/ 5W1H	Multi quotes ("")
Contains a number	Contains "reason"	Exclamation pt. (!)
Contains an acronym	Contains "why"	Parenthesis ()
Contains contraction	Contains "just"	Asterisk (*)
	Contains "this"	Question mark (?)
	Contains "one"	Colon (:)
		Semicolon (;)
		Money ($)

4. Feature Analysis

The statistical analysis results in Table 2 show that all 30 features exhibit definitive different occurrence rates between clickbait and non-clickbait confirming their usefulness for classifying headlines. Several feature statistical differences stand out more than others signifying potential top classifiers. The features *"Ratio words begin w/uppercase", " Ratio of stop-words ", "Begins w/ Number", "Contains possessive/pronoun" and "Contains demonstrative"* have significant occurrence rate differences between clickbait and non-clickbait headlines.

Table 2. Feature statistics.

	Headline Averages							
Type	**Word Count**	**Word Length**	**Upper Case Ratio**	**Stop Word Ratio**	**F Score**	**LIX**	**RIX**	**CL Score**
Clickbait	10.0	4.7	1.0	0.0	61.7	29.4	1.9	8.4
Non–Clickbait	8.4	5.3	0.5	0.2	78.5	40.5	2.6	11.7

	Percent Begin w/			**Percent of Headlines Containing**				
Type	**5W1H**	**Number**	**Det/Sep**	**Number**	**Det/Sup**	**Pronoun**	**Contract**	**Demons**
Clickbait	7.9%	37.4%	19.3%	46.0%	65.4%	58.7%	23.2%	25.9%
Non-Clickbait	0.4%	4.9%	3.2%	24.0%	20.8%	8.8%	7.2%	1.1%

	Percent of Headlines Containing							
Type	**"**	**!**	**( )**	*****	**?**	**:**	**;**	**$**
Clickbait	4.2%	0.2%	0.4%	0.1%	0.3%	2.3%	0.0%	0.3%
Non-Clickbait	0.6%	0.1%	0.2%	0.0%	0.7%	4.4%	1.4%	1.3%

	Percent of Headlines Containing					
Type	**"reason"**	**"why"**	**"just"**	**"this"**	**"one"**	**Acronym**
Clickbait	0.2%	1.3%	2.4%	11.2%	1.9%	6.3%
Non-Clickbait	0.0%	0.0%	0.2%	0.3%	1.2%	15.5%

The stacked bar chart in Figure 1 shows the normalized occurrence rate of the 22 binary features in clickbait and non-clickbait headlines. Features on the right with normalized rates above 0.5 are more associated with clickbait headlines. The higher the rate the higher the association with clickbait headlines. The opposite is true on the left. Features on the left with normalized rates below 0.5 are more associated with non-clickbait headlines. The lower the rate the higher the association with non-clickbait headlines.

The eight continuous numeric value features (ratio and count features) were transformed into binary values using discretization. The stacked bar chart in Figure 2 shows the normalized occurrence rate in clickbait and non-clickbait headlines for these eight transformed features. The two features *"Uppercase2"* (headlines with all words beginning in uppercase) and *"Word Count2"* have a higher rate of association with clickbait headlines. The other six features are more strongly associated with non-clickbait headlines.

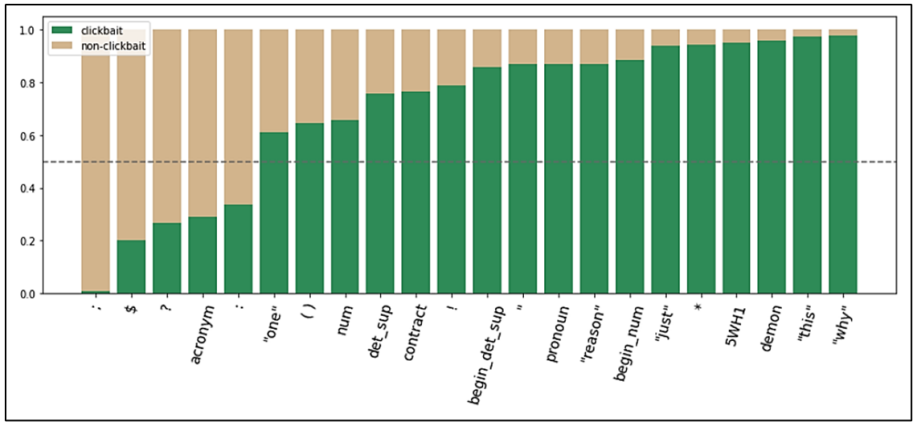

Figure 1. Stacked Bar Chart of Binary Feature Normalized Occurrence Rates in Clickbait Headlines.

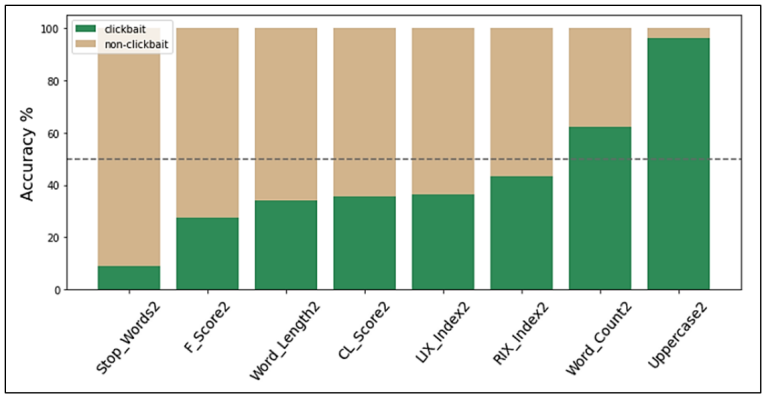

Figure 2. Stacked Bar Chart of Feature Normalized Occurrence Rates in Clickbait Headlines.

Figure 3 contains the correlation matrix of all 30 features. The Correlation matrix utilizes the Pearson coefficient of correlation between each of the features. The Pearson coefficient of correlation is a linear correlation with a range of -1 to $+1$. A value of -1 signifies a strong negative correlation while a $+1$ indicates a strong positive correlation. The matrix is also color coded with shades of red being associated with positive (+) coefficients and shades of blue with negative ($-$). The darker the color, the higher the correlation.

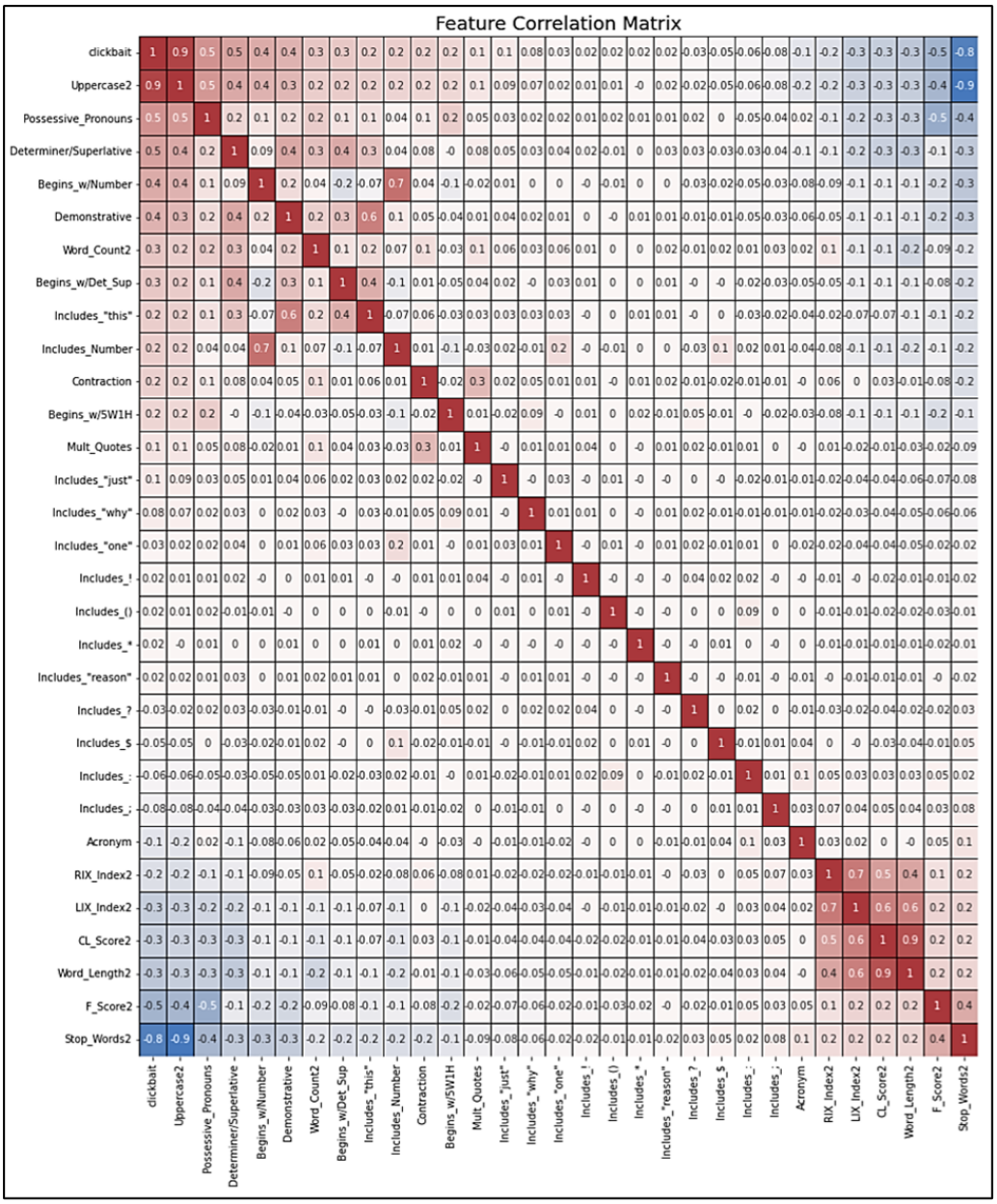

Figure 3. Feature Correlation Matrix.

Overall, the feature correlation +/− groupings in the matrix match the predicted clickbait/non-clickbait associations in the stacked bar charts. There are two features that have very high correlation values. The feature *"Uppercase2"* has a coefficient of correlation of 0.9, which is an almost perfect correlation with the clickbait target classification. On the other end of the spectrum, the feature *"Stop Words2"* has a coefficient of correlation of −0.8, which indicates a very strong correlation with a non-clickbait target classification. Table 3 contains a listing of the 15 features with the highest correlation values.

Table 3. Top-15 features.

The Top-10 Features Correlated with "Clickbait" Headlines	
(1) Uppercase2	0.93
(2) Possessive Pronouns	0.53
(3) Determiner/Superlative	0.45
(4) Begins w/Number	0.40
(5) Demonstrative	0.36
(6) Word Count2	0.27
(7) Begins w/Det Sup	0.26
(8) Includes "this"	0.23
(9) Includes Number	0.23
(10) Contraction	0.22
The Top-5 features Correlated with "Non-Clickbait" Headlines	
(1) Stop Words2	−0.83
(2) F Score2	−0.46
(3) Word Length2	−0.33
(4) CL Score2	−0.29
(5) LIX Index2	−0.29

5. Modeling

This section describes the modeling approach used to classify clickbait and non-clickbait headlines. The classification modeling was performed in Python [22] using the scikit-learn machine learning library [23]. The data analysis and modeling were conducted in Jupyter notebook, an open-source web application that allows for interactive data science and scientific computing using the Anaconda distribution.

5.1. Individual Modeling

Six individual classification models are tested: decision tree, logistic regression, naïve Bayes, support vector machine (svm), k-nearest neighbor (knn) and gradient-boosted decision tree (GBDT). The dataset is randomly split 80:20 into 25,600 training headlines and 6400 test headlines. In order to optimize the performance of the six machine learning models, we conducted a thorough search for the best hyperparameters. The hyperparameters of each model are selected through a randomized search, which is a probabilistic method for hyperparameter tuning. The randomized search was conducted for 100 iterations for each model and the best hyperparameters were chosen based on the highest performance metric score on the validation set. The hyperparameters selected for each machine learning model are displayed in Table 4.

Table 4. Hyperparameters selected for each machine learning algorithm.

Model	Selected Hyperparameters
Decision Tree	Maximum Depth = 10
Logistic Regression	Regularization Parameter = 0.01
Naive Bayes	None
Support Vector Machine	Regularization Parameter = 0.1, Kernel = Polynomial
K-Nearest Neighbor	Number of Neighbors = 5
Gradient Boosting	Learning Rate = 0.1, Maximum Depth = 5

After selecting the hyperparameters for each of the six machine learning models, the next step is to evaluate their effectiveness in categorizing clickbait headlines. The headline dataset was partitioned into two subsets—a training set comprising 25,600 headlines and a test set comprising 6400 headlines, using an 80:20 ratio. Before evaluating the models, it is important to understand the key metrics used to assess the models' performance. The selected evaluation metrics are accuracy, precision and recall.

1. **Accuracy** measures the proportion of correct predictions out of all predictions made and is calculated as:

$$\textbf{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (5)$$

where TP (true positive) is the number of actual clickbait headlines that were correctly classified as clickbait, TN (true negative) is the number of actual non-clickbait headlines that were correctly classified as non-clickbait, FP (false positive) is the number of actual non-clickbait headlines that were incorrectly classified as clickbait and FN (false negative) is the number of actual clickbait headlines that were incorrectly classified as non-clickbait.

2. **Precision** measures the proportion of correct clickbait predictions out of all predictions made as clickbait and is calculated as:

$$\textbf{Precision} = \frac{TP}{TP + FP} \tag{6}$$

3. **Recall** measures the proportion of actual clickbait headlines that were correctly classified as clickbait out of all actual clickbait headlines and is calculated as:

$$\textbf{Recall} = \frac{TP}{TP + FN} \tag{7}$$

In addition to the full 30 classification features, the models were trained and tested using less than the 30 features (i.e., Top-25/20/15/10/5/2 features). The goal is to find the simplest model that produces the best accuracy utilizing the least number of features. Table 5 shows the validation performance of each of the models across the different combined feature sets. The results in Table 5 show all six models produced an accuracy, precision and recall greater than 0.96. The *SVM* and *GBDT* models produced the best results with an accuracy of 0.98, a precision of 0.98 and a recall of 0.97. Furthermore, the performance of these two models did not significantly change between using the full 30 features and only using the top-15 features. When less than 15 features were used the performance of all the models started dropping slightly.

5.2. Ensemble Modeling

The top-5 most accurate models (*SVM, GBDT, decision tree, logistic regression and KNN*) were combined into an ensemble model. The ensemble model was run 10× with random 80:20 training/test sets using only the top–15 features. Majority voting between the models was used on each run to produce the accuracy results. The average accuracy from the 10 runs was 0.976, with a standard deviation of 0.001. The average standard deviation was very low, indicating a very tight grouping around the average. The average accuracy for the combined models did not improve over the accuracy of the individual top models.

One explanation for the lack of improvement in accuracy for the ensemble model is that the model may produce closely matching sets of false-positive and false-negative headlines. When analyzed, over 75% of the false-positive and over 65% of the false-negative results are the same headlines in all five models. Examining the false-positive results, the "*Uppercase2*" feature appears to be the primary culprit. However, this feature is also the top feature for classifying headlines as clickbait. Similarly, the "*Stop Words2*" feature appears to be the primary cause of the misclassifications resulting in false-negative results. Yet, this feature is the top feature for classifying headlines as non-clickbait. The conclusion from this analysis is that there are no adjustments that can be made to the ensemble models and feature set that would improve the accuracy more.

Table 5. Model Validation Performance (Accuracy, Precision, Recall).

Decision Tree							
	Number of Classification Features						
	30 Feat	**25 Feat**	**20 Feat**	**15 Feat**	**10 Feat**	**5 Feat**	**2 Feat**
Accuracy	0.973	0.973	0.974	0.973	0.968	0.967	0.966
Precision	0.976	0.975	0.976	0.976	0.978	0.942	0.944
Recall	0.971	0.971	0.973	0.970	0.959	0.995	0.993
Logistic Regression							
	Number of Classification features						
	30 Feat	**25 Feat**	**20 Feat**	**15 Feat**	**10 Feat**	**5 Feat**	**2 Feat**
Accuracy	0.975	0.975	0.975	0.973	0.970	0.967	0.966
Precision	0.978	0.978	0.978	0.969	0.964	0.943	0.944
Recall	0.973	0.973	0.973	0.978	0.976	0.994	0.993
Naïve Bayes							
	Number of Classification features						
	30 Feat	**25 Feat**	**20 Feat**	**15 Feat**	**10 Feat**	**5 Feat**	**2 Feat**
Accuracy	0.960	0.959	0.958	0.960	0.960	0.963	0.966
Precision	0.958	0.958	0.956	0.955	0.959	0.936	0.944
Recall	0.962	0.961	0.961	0.967	0.962	0.995	0.993
Support Vector Machine							
	Number of Classification features						
	30 Feat	**25 Feat**	**20 Feat**	**15 Feat**	**10 Feat**	**5 Feat**	**2 Feat**
Accuracy	**0.975**	**0.975**	**0.976**	**0.976**	0.969	0.967	0.966
Precision	0.982	0.981	0.979	0.979	0.967	0.942	0.944
Recall	0.969	0.969	0.973	0.974	0.971	0.995	0.993
k-Nearest Neighbor							
	Number of Classification features						
	30 Feat	**25 Feat**	**20 Feat**	**15 Feat**	**10 Feat**	**5 Feat**	**2 Feat**
Accuracy	0.973	0.970	0.968	0.974	0.970	0.967	0.966
Precision	0.976	0.974	0.969	0.978	0.966	0.942	0.944
Recall	0.971	0.967	0.967	0.970	0.974	0.995	0.993
Gradient-Boosted Decision Tree							
	Number of Classification features						
	30 Feat	**25 Feat**	**20 Feat**	**15 Feat**	**10 Feat**	**5 Feat**	**2 Feat**
Accuracy	**0.976**	**0.976**	**0.977**	**0.975**	0.968	0.967	0.966
Precision	0.979	0.979	0.979	0.979	0.966	0.942	0.944
Recall	0.974	0.974	0.974	0.972	0.971	0.995	0.993

5.3. Factor Analysis Modeling

In a final attempt to improve the classification accuracy, exploratory factor analysis [24,25] was conducted on the features and then the independent and ensemble models were re-ran with the combined factors. Exploratory factor analysis is a linear statistical method used to summarize a large set of features into smaller variables called factors. To confirm that factor analysis was indeed feasible for the given headline features, the Bartlett sphericity test and the Kaiser–Meyer–Olkin (KMO) test were used.

The Bartlett sphericity test checks whether or not the features (observed variables) are intercorrelated by comparing the observed correlation matrix and the identity matrix. If the two are not the same, the test is significant. For the test of our feature set, the Chi-Square was 26,0470.01 and the *p*-value was 0, signifying that factor analysis is feasible.

The Kaiser–Meyer–Olkin (KMO) [10] test estimates the proportion of variance among all the observed variables. KMO values range between 0 and 1 with a value of 0.6 or more indicating factor analysis is feasible. For the test of our feature set, the KMO value was 0.73, again indicating factor analysis is feasible.

Next, the Kaiser criterion [26] was used to determine the number of factors. The Kaiser criterion is an analytical approach, which is based on the selection of factors that explain a more significant proportion of variance. The eigenvalue is used as an index for the variance as a portion of the total variance and it indicates how good a component is as a summary of the data. An eigenvalue of means that the factor contains the same amount of information as a single feature. Generally, an eigenvalue greater than 1 is considered a good selection criterion for a factor. The scree plot in Figure 4, which is a plot of eigenvalues

and feature/factor numbers, is a graphical representation of the Kaiser criterion. The "elbow" in the curve on the scree plot, just before the line flattens out, corresponds to the number of factors to select. The "elbow" in the scree plot curve indicates 5 factors as the optimum choice.

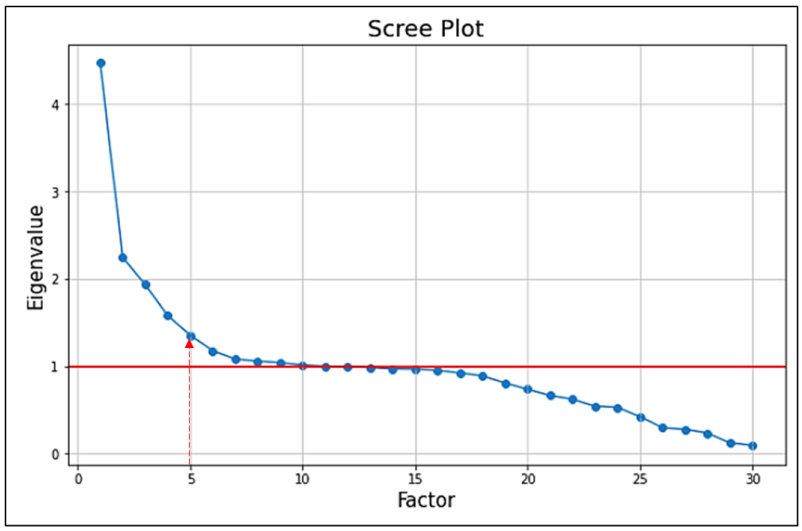

Figure 4. Scree Plot.

Python's Factor Analyzer with oblique rotation was used to extract the five factors and produce a factor loading matrix in Table 6. Factor loading indicates how well a factor is able to explain a feature. A low factor loading indicates the feature does not belong to the factor. A high factor loading indicates the item belongs to the factor. A factor load of 0.30 was used as a cutoff for the pairing of features to factors. Not all of the 30 features are rated high enough to be included in a factor. The features included in the five factors are:

- Factor-1: *"Word Length2", "LIX Index2", "RIX Index2" and "CL Score2"*
- Factor-2: *"Begins w/5W1H", "Possessive Pronouns", "Uppercase2", "Stop Words2" and "F Score2"*
- Factor-3: *"Begins w/Det Sup", "Determiner/Superlative", "Demonstrative" and "Includes "this""*
- Factor-4: *"Begins w/Number" and "Includes Number"*
- Factor-5: *"Contraction", "Mult Quotes", "Word Count2" and "RIX Index2"*

The new factor values were computed by summing the product of the feature values by its corresponding factor load for each instance (headline) in the dataset. Each new factor was then normalized across the dataset.

The factor-based headline dataset was randomly split 80:20 into training and test sets. The six individual classification models were trained and tested using the five factors. Table 7 shows the validation performance of each of the models. The accuracy of the factor models was no better than the results from the feature models, with an accuracy of 0.98.

Next, the ensemble model was run 10× with random 80:20 training/test sets using the five factors. The average accuracy from the 10 runs was 0.975, with a standard deviation of 0.002. The average standard deviation was very low, indicating a very tight grouping around the average. The average accuracy for the combined models again did not improve over the accuracy of 0.976 for the individual top models.

Table 6. Five Factor Loadings Matrix.

Feature	Factors F1	F2	F3	F4	F5
Begins w/5W1H	−0.09	0.36	−0.21	−0.24	−0.09
Begins w/Number	0.02	0.13	−0.02	0.87	−0.04
Begins w/Det Sup	−0.06	0.08	0.48	−0.22	0.04
Includes Number	−0.08	-0.08	0.00	0.81	0.05
Determiner/Superlative	−0.21	0.13	0.43	−0.02	0.18
Possessive Pronouns	−0.16	0.60	−0.07	−0.13	0.06
Contraction	0.05	0.25	−0.09	−0.03	0.35
vDemonstrative	0.01	0.04	0.76	0.11	−0.03
Includes "reason"	−0.01	0.01	0.02	0.00	0.03
Includes "why"	−0.02	0.10	−0.03	−0.04	0.02
Includes "just"	−0.03	0.07	0.02	0.00	0.03
Includes "this"	0.02	−0.01	0.74	−0.10	−0.01
Includes "one"	−0.06	−0.03	0.03	0.08	0.07
Acronym	−0.05	−0.11	−0.07	−0.04	0.08
Mult Quotes	−0.02	0.15	−0.09	−0.07	0.32
Includes !	−0.01	0.02	−0.01	−0.01	0.03
Includes ()	−0.02	0.02	−0.02	−0.02	0.01
Includes *	−0.01	0.01	0.00	−0.01	−0.01
Includes ?	−0.05	0.00	−0.02	−0.05	0.00
Includes :	0.01	−0.05	−0.04	−0.01	0.07
Includes ;	0.02	−0.08	−0.02	0.01	0.07
Includes $	−0.05	−0.08	0.00	0.06	0.03
Word Count2	−0.15	0.09	0.15	0.01	0.44
Word Length2	0.91	0.01	−0.01	−0.03	−0.13
Uppercase2	−0.01	0.90	0.06	0.06	0.03
Stop Words2	−0.07	−0.85	−0.04	−0.06	0.01
F Score2	0.08	−0.54	0.03	0.03	0.06
LIX Index2	0.74	−0.07	−0.02	0.00	0.18
RIX Index2	0.56	−0.06	0.01	0.00	0.36
CL Score2	0.89	0.01	0.01	−0.02	−0.02

Table 7. Factor Model Validation Performance.

DECISION TREE	Accuracy: Precision: Recall:	0.973 0.977 0.971
LOGISTIC REGRESSION	Accuracy: Precision: Recall:	0.969 0.968 0.972
NAÏVE BAYES	Accuracy: Precision: Recall:	0.788 0.747 0.878
SVM	Accuracy: Precision: Recall:	**0.975** 0.974 0.976
kNN	Accuracy: Precision: Recall:	0.969 0.965 0.975
GBDT	Accuracy: Precision: Recall:	**0.976** 0. 978 0. 974

6. Accuracy Comparisons with Similar Studies

In Table 8, we compare our model accuracy results with similar clickbait studies that employed similar machine learning techniques on similar datasets. Notice our models, utilizing only the top-15 semantic features, show a marked improvement in accuracy over all the non-neural-net model studies in Table 8. The neural network model studies of Kumar, V. et al. [14] and Anand, A. et al. [27] utilize a different feature approach from the semantic features of the other classification models. Still, our model's performance matches or beats the neural network models. The model's performance could be improved using more data and further feature engineering.

Table 8. Model Accuracy Comparisons to Related Works.

Related Work	Model	Accuracy
Biyani, P., et al. [4]	Gradient-Boosted Decision Trees (GBDT)	76%
Chakraborty, A., et al. [7]	Decision Tree	90%
	Random Forest	92%
	Support Vector Machine (SVM)	93%
Salerno, A. [13]	Random Forest	91%
	Logistic Regression	93%
	Naïve Bayes	93%
	Support Vector Machine (SVM)	93%
Pujahari, A., et al. [14]	Decision Tree	92%
	Random Forest	94%
	Support Vector Machine (SVM)	97%
Kumar, V., et al. [15]	Bi-Directional Long Short-Term Memory (BiLSTM) Recurrent Neural Network	83%
Anand, A., et al. [16]	Bi-Directional Long Short-Term Memory (BiLSTM) Recurrent Neural Network	98%
Our Models (w/ 15 features)	**Decision Tree**	**97%**
	Logistic Regression	**97%**
	Naïve Baye	**96%**
	Support Vector Machine (SVM)	**98%**
	k-Nearest Neighbor (kNN)	**97%**
	Gradient-Boosted Decision Trees (GBDT)	**98%**

7. Conclusions and Future Directions

This paper presented an effective method for categorizing clickbait and non-clickbait headlines using semantic analysis and machine learning techniques. It was shown that high accuracy in classifying clickbait headlines could be achieved by investigating thirty unique semantic features and exploring six different machine learning classification algorithms, both individually and as ensembles. The top models, including the support vector machine and gradient-boosted decision tree algorithm, achieved an accuracy of 98%. Furthermore, the results indicated that even with only two key semantic features, an accuracy of 97% could be achieved. These results outperformed previous models in the literature. However, it is important to note that the dataset used in this study was collected from a limited set of news websites and it is possible that the results may not generalize to other sources. Additionally, this study only focused on clickbait headlines and not the content of the articles. In future work, it would be interesting to expand the study to include the article content and evaluate the model's performance on a larger dataset. Furthermore, the proposed models could be used as a template for developing a practical application to automatically classify clickbait headlines and potentially deploy a real clickbait detection extension for a web browser such as Safari.

Author Contributions: Conceptualization, M.B. and M.A.-k.; Data curation, M.B. and M.A.-k.; Formal analysis, M.B. and M.A.-k.; Investigation, A.A.B., M.B. and M.A.-k.; Methodology, M.B. and M.A.-k.; Project administration, M.B., M.A.-k. and A.A.B.; Software, M.B. and M.A.-k.; Supervision, M.A.-k. and A.A.B.; Validation, M.B., M.A.-k. and A.A.B.; Visualization, M.B., M.A.-k. and A.A.B.; writing—original draft preparation, M.B.; Writing—review and editing, M.B., M.A.-k. and A.A.B.; funding acquisition, A.A.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research is partially funded by Lewis University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: https://www.kaggle.com/datasets/amananandrai/clickbait-dataset, accessed on 3 August 2022.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Chakraborty, A.; Paranjape, B.; Kakarla, S.; Ganguly, N. Stop clickbait: Detecting and preventing clickbaits in online news media. In Proceedings of the 2016 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM), Davis, CA, USA, 18–21 August 2016; pp. 9–16.
2. Potthast, M.; Köpsel, S.; Stein, B.; Hagen, M. Clickbait detection. In Proceedings of the European conference on information retrieval, Padua, Italy, 20–23 March 2016; pp. 810–817.
3. Pujahari, A.; Sisodia, D.S. Clickbait detection using multiple categorisation techniques. *J. Inf. Sci.* **2021**, *47*, 118–128. [CrossRef]
4. Christin, A. Counting clicks: Quantification and variation in web journalism in the United States and France. *Am. J. Sociol.* **2018**, *123*, 1382–1415. [CrossRef]
5. Al Bataineh, A.; Kaur, D. Immunocomputing-based approach for optimizing the topologies of LSTM networks. *IEEE Access* **2021**, *9*, 78993–79004. [CrossRef]
6. Al Bataineh, A.; Kaur, D.; Jalali, S.M.J. Multi-Layer Perceptron Training Optimization Using Nature Inspired Computing. *IEEE Access* **2022**, *10*, 36963–36977. [CrossRef]
7. Razaque, A.; Alotaibi, B.; Alotaibi, M.; Hussain, S.; Alotaibi, A.; Jotsov, V. Clickbait Detection Using Deep Recurrent Neural Network. *Appl. Sci.* **2022**, *12*, 504. [CrossRef]
8. Shang, L.; Zhang, D.Y.; Wang, M.; Lai, S.; Wang, D. Towards reliable online clickbait video detection: A content-agnostic approach. *Knowl. Based Syst.* **2019**, *182*, 104851. [CrossRef]
9. Liao, F.; Zhuo, H.H.; Huang, X.; Zhang, Y. Federated hierarchical hybrid networks for clickbait detection. *arXiv* **2019**, arXiv:1906.00638.
10. Agrawal, A. Clickbait detection using deep learning. In Proceedings of the 2016 2nd international conference on next generation computing technologies (NGCT), Dehradun, India, 14–16 October 2016; pp. 268–272.
11. Setlur, A.R. Semi-Supervised Confidence Network aided Gated Attention based Recurrent Neural Network for Clickbait Detection. *arXiv* **2018**, arXiv:1811.01355.
12. Fakhruzzaman, M.N.; Jannah, S.Z.; Ningrum, R.A.; Fahmiyah, I. Clickbait Headline Detection in Indonesian News Sites using Multilingual Bidirectional Encoder Representations from Transformers (M-BERT). *arXiv* **2021**, arXiv:2102.01497.
13. Thomas, P. Clickbait identification using neural networks. *arXiv* **2017**, arXiv:1710.08721.
14. Kumar, V.; Khattar, D.; Gairola, S.; Kumar Lal, Y.; Varma, V. Identifying clickbait: A multi-strategy approach using neural networks. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; pp. 1225–1228.
15. Cao, X.; Le, T. Machine learning based detection of clickbait posts in social media. *arXiv* **2017**, arXiv:1710.01977.
16. Amananandrai. Clickbait Dataset. 2020. Available online: https://www.kaggle.com/datasets/amananandrai/clickbait-dataset (accessed on 3 August 2022).
17. Blom, J.N.; Hansen, K.R. Click bait: Forward-reference as lure in online news headlines. *J. Pragmat.* **2015**, *76*, 87–100. [CrossRef]
18. Biyani, P.; Tsioutsiouliklis, K.; Blackmer, J. 8 amazing secrets for getting more clicks: Detecting clickbaits in news streams using article informality. In Proceedings of the Thirtieth AAAI conference on artificial intelligence, Phoenix, AZ, USA, 12–17 February 2016.
19. Coleman, M.; Liau, T.L. A computer readability formula designed for machine scoring. *J. Appl. Psychol.* **1975**, *60*, 283. [CrossRef]
20. Anderson, J. Lix and rix: Variations on a little-known readability index. *J. Read.* **1983**, *26*, 490–496.
21. Heylighen, F.; Dewaele, J.M. *Formality of Language: Definition, Measurement and Behavioral Determinants*; Interner Bericht, Center "Leo Apostel", Vrije Universiteit Brüssel: Brussel, Belgium, 1999; Volume 4.
22. VanRossum, G.; Drake, F.L. *The Python Language Reference*; Python Software Foundation: Amsterdam, The Netherlands, 2010.
23. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
24. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An Introduction to Statistical Learning*; Springer: Berlin/Heidelberg, Germany, 2013; Volume 112.
25. Everitt, B.S. *Multivariable Modeling and Multivariate Analysis for the Behavioral Sciences*; CRC Press: Boca Raton, FL, USA, 2009.
26. Braeken, J.; Van Assen, M.A. An empirical Kaiser criterion. *Psychol. Methods* **2017**, *22*, 450. [CrossRef] [PubMed]
27. Anand, A.; Chakraborty, T.; Park, N. We used neural networks to detect clickbaits: You won't believe what happened next! In Proceedings of the European Conference on Information Retrieval, Aberdeen, UK, 8–13 April 2017; pp. 541–547.