

Detecting and Categorization of Click Baits

Prof. Sainath Patil

Department of Information Technology
Vidyavardhini's College of Engineering and Technology
Vasai, India

Mayur Koul

Department of Information Technology
Vidyavardhini's College of Engineering and Technology
Vasai, India

Harikrishan Chauhan

Department of Information Technology
Vidyavardhini's College of Engineering and Technology
Vasai, India

Prachi Patil

Department of Information Technology
Vidyavardhini's College of Engineering and Technology
Vasai, India

Abstract—Clickbaiting is a growing phenomena on the internet, and it is defined as a method of exploiting cognitive biases to attract online viewership, that is, to attract clicks. The articles behind clickbaits are usually uninformative, and besides contributing to an overall decline in journalistic integrity, clickbaits spread misinformation, often by making shocking implications, only to backtrack on those claims in the article. Examples of clickbait titles are: 10 ways the expat life is like a continual espresso buzz, 5 incredible Italian dishes you haven't tried before, What India's microloan meltdown taught one entrepreneur. Clickbaiting can come in many forms, like advertisements, videos, and such, but the matter is that websites are highly incentive to publish clickbait articles because they're cheap to produce and can generate revenue. The main motivation in trying to spot clickbait is to separate out potential sources of misinformation on the net. With the event of online advertisements, clickbait spread wider and wider. Clickbait dissatisfies users because the article content does not match their expectation. A convolutional neural network is beneficial for clickbait detection, since it utilizes pre-trained Word2Vec to know the headlines semantically, and employs different kernels to search out various characteristics of the headlines. However, different types of articles tend to use alternative ways to draw users attention, and a pre-trained Word2Vec model cannot distinguish these alternative ways. We propose a completely unique approach considering all information found during a social media post. We train a bidirectional Long Short-Term Memory (LSTM) with an attention mechanism to learn the extent to which a word contributes to the posts' clickbait score in a differential manner.

Index Terms—Clickbait, Convolutional Neural Network (CNN), Deep Learning.

I. INTRODUCTION

Nowadays on internet a varied type of content is found which is entertaining for the users. Earlier users used to stick to a particular trusted site for their knowledge gaining, but according to current trends it is observed that the users tend to select any of the site as their source of information. For being in the competition of this cruel world the news sites have taken a digital approach to it. They have started generating the revenue through their digital front by providing advertisements on their portal. However, since the same information is available via multiple sources, no comment will be made on the preference of the reader. To lure in additional readers and increase the quantity of clicks on their content, subsequently increasing their agencies revenue, writers have begun adopting a new technique clickbait. The concept of clickbait is formalized as something to encourage readers to click on hyperlinks supported snippets of information accompanying it,

especially when those links result in content of dubious value or interest. Clickbaiting is that the intentional act of over promising or purposely misrepresenting in a headline, on social media, in an image, or some combination what will be expected while reading a story on the net. It is designed to create and, consequently, capitalize on the Loewenstein information gap [11]. Most of the times the links provided in such fake headlines often lead to dummy or malicious pages whose original content has already been reviewed. However

This Cat Was Trying To Get The Attention Of Locals, But What Someone Finds With Her? Unbelievable

Little Girl Tries To Wake Her Brother Up. What He Did Was Truly Unbelievable!

Wait! Don't Put Butter on That Grilled Cheese Sandwich. Do THIS Instead! (You Can Thank Us Later.)

Fig. 1. Some Examples of Clickbait.

for many of the part their method of detecting the clickbait in manual, the user running those twitter accounts themselves read and classify the tweet as clickbait or not for the benefit of the user. There are numerous reasons behind the creation of such clickbait articles with genuine sources identifying and acknowledging the claim. Reis et al [7] studied around 69,000 headlines from four international media houses since 2014. They analyzed the headlines and detected extreme polarity in the sentimental value of it, thus resulting in more popularity and generating interest as well as curiosity in the minds of the user. A user reads an article only after reading and generating interest in the headline, so it all depends on how the user or customer perceives the headline and the article of which a detailed study was done by Digirolamo and Hintzman [8]. By simply drawing attention to certain details or facts one can easily lure the user into clicking the link, because it generates a sense of existing knowledge been activated in their brains. By its choice of phrasing and choice of words, a headline can



Fig. 2. Comments that were found in clickbait videos. The users frustration is apparent (we omit users names for ethical reasons).

influence ones mindset and mislead them into perceiving the content with some other reference consistent with the headline as shown by Dooling and Lachman[9]. As discussed earlier the prominent explanation is the frequently cited Loewenstein information gap theory [10]. It generates emotional consequences as we perceive wrongly and created gap between what we know and what we want to know. Our engine is built on three components. The first leverages neural networks for sequential modelling of text. Article title is represented as a sequence of word vectors and every word of the title is further converted into character level embeddings. These features function input to a bidirectional LSTM model. An affixed attention layer allows the network to treat each word within the title in an exceedingly differential manner. The subsequent component focuses on the similarity between the article title and its actual content. For this, we generate Doc2Vec embeddings for the pair and act as input for a Siamese net, projecting them into a highly structured space whose geometry reflects complex semantic relationships. Lastly the similarity of the image attached with is quantified. Finally, every components output is integrated and served as an input to the fully connected layer to get an output value of the work/task in hand.

II. RELATED WORK

Clickbait has been a subject of research twice by linguists and twice by computer science research teams. Amongst linguists, Bram Vijgen [11] studied listicles which are articles containing a list of things. Listicles are one of the major types of clickbaits. The titles like 16 Cancer Causing Foods You Probably Eat Every Day or 38 Celebrities You Didn't Know Passed are some examples which are taken from our own compiled data-set. The authors studied around 700 listicles by BuzzFeed. They found that the titles share a very homogeneous structure: 85 percent of them starting with a cardinal number the number of items in the list while all articles contain the number in some place or the other. Second was Blom and Hansen [12], who studied usage of forward reference in 2000 random headlines from a Danish news website. Forward reference is utilized to create an information gap by giving only a teasing headline and luring the user to click the title. Some examples include This shocking news will blow your mind or What He did next shocked everyone. In the aforementioned examples This and He are the forward references to some entities which are not disclosed, enticing the user to click to find out. They found that these forward references are mostly made up of definite articles, adverbs and personal and demonstrative

pronouns. The teaser message had some basic text and dictionary features. The linked web page features included text and readability features, while the meta information had features related to the tweets themselves. These features were fed into a supervised classification mechanism which achieved 0.79 ROC-AUC at 0.76 precision and 0.76 recall. They found that features from category one alone outperformed all other categories, with character n-gram and word 1-gram features contributing the most as they are known to capture writing styles. Many studies were conducted by observing the behaviors of internet users and making inferences. In their study, Hermann et al. gathered data by using an IP based data gathering method and processed this data by using semi supervised machine learning method, determining that 87 of the users use the internet with a certain pattern. Mngen and Kaya [13] suggested the activity recommendation system on the social networks by using patterns. Semantic Web is meaningful data extraction from the web and has been frequently done for the last 15 years [14]. Toma et al. [15] performed Semantic Web operations through big data methods. One of the most important areas of Semantic Web is natural language processing. In general, Clickbait news have been detected with natural language processing methods. One of the first studies on Clickbait was done by Chakraborty et al [1]. While studying this more than 6 different examinations were done on news sources. 15,000 news were gathered from these news sources. While 7,500 of those were labeled as true news, the remaining 7,500 of reports were labeled as Clickbait news. A clickbait detection model has been created which takes news headlines as an input. During this, it is aimed to get a unique way to examine the news sites for eliminating clickbait headlines and without clicking show the news content. With an extension created for applications programme, news was automatically checked, and news content was shown without clicking. The clickbait problem is additionally identified by prior work that proposes tools for alleviating the problem in various web portals. Specifically, Chen et al. [16] provide useful information regarding the clickbait problem and future directions for tackling the matter using SVM and Naive Bayes approaches. Rony et al. [17] analyze 1.67M posts on Facebook so as to know the extent and impact of the clickbait problem still as users engagement. For detecting clickbaits, they propose the use of sub-word embeddings with a linear classifier. Potthast et al. [18] target the Twitter platform where they suggest the utilization of Random Forests for distinguishing tweets that contain clickbait content. Furthermore, Chakraborty et al. [1] propose the utilization of SVMs in conjunction with a browser add-on for offering a detection system to end-users of reports articles. Biyani et al. [19] recommends the usage of Gradient Boosted Decision Trees for detecting the clickbait in news content. They also demonstrate that the degree of informality in the content of the landing page can help in discerning clickbait news articles. To the most effective of our knowledge, Anand et al. [20] is that the first work that implies the use of deep learning techniques for mitigating the clickbait problem. Specifically, they propose the use of Recurrent Neural Networks in conjunction with word2vec embeddings for identifying clickbait news articles. In contrast, our method gives

up the time consuming and painstaking task of hand-created features. Instead we utilize deep learning to seek out features. We get excellent results utilizing this system, which might be because deep learning learns features on its own and it might be coming up with new and un-thought of features. Also, the information set collected by us comes from various sources which helps us in developing a more generalized model that is not constrained by the type of social media platform. The task of automating clickbait detection has risen to prominence fairly recently. Initial attempts for the same have worked on news headlines, and heavy feature engineering for the actual data set. [3]'s work is one in all the earliest pieces of literature available within the field, specializing an aggregation of news headlines from previously categorized clickbait and non-clickbait sources. Apart from defining different forms of clickbait, they emphasize on the presence of language peculiarities exploited by writers for this purpose. These include qualitative informality metrics and use of forward references in the title to keep the reader on the hook. The primary instance of detecting clickbait across social media can be traced to [15], hand-crafting linguistic features, including a reference dictionary of clickbait phrases, over a data set of crowdsourced tweets [14]. However, [4] argued that work done specifically for Twitter had to be expanded since clickbait was available throughout the Internet, and not just social networks. It was not until [1] that neural networks were tried out for the task as the authors used the same news data set as [4] to develop a deep learning based model to detect clickbait. They used distributional semantics to represent article titles, and Bidirectional LSTM (BiLSTM) to model sequential data and its dependencies. Since then, [18] has also experimented with Twitter data [14] deploying a BiLSTM for each of the textual features (posttext, target-title, target-paragraphs, target description, target keywords, post-time) available in the corpus, and finally concatenating the dense output layers of the network before forwarding it to a fully connected layer. Since it was proposed in [2], here an attention mechanism has been used to carry out several text classification tasks, which includes malicious news and sentiment analysis [20] used a self-attentive Bidirectional Gated Recurrent Units (BiGRU) to infer the importance of tweet tokens in predicting the annotation distribution of the task. One common point in all the approaches yet has been the use of only textual features available in the data set. Our model not only incorporates textual features, modelled using BiLSTM and augmented with an attention mechanism, but also considers related images for the task. Curiosity gap is the psychological phenomenon which the clickbait exploits as a cognitive basis of the earliest study. To increase the chances of someone clicking on the link clickbait holds a strong factor of using vague and interesting language ultimately to intrigue the user [5]. This has also been studied in linguistics. There have been studies on listicles, or articles that are simply lists of things, that are sensational and clickbait [7]. Additionally, clickbait articles tend to have forward references like this or these [1]. These findings have identified tell tale characteristics

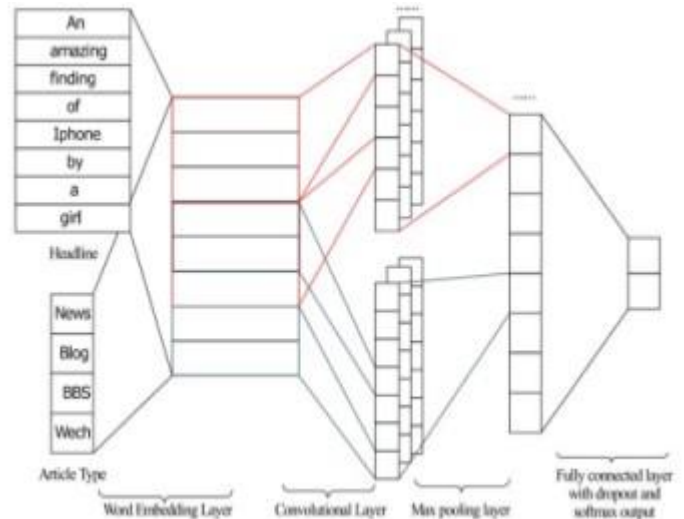


Fig. 3. The structure of the clickbait convolutional neural network.

of Fig. 3. The structure of the clickbait convolutional neural network. Clickbait articles, but a model that extracts only these features would not be robust. The features need to be more nuanced to avoid flagging non-clickbait articles. Recently, machine learning approaches to clickbait detection have been proposed. Potthast et al. (2016) was the first one to publish a self efficient clickbait detection model which had a unique feature of extraction from the post text, the metadata and the various raw webpage and used LR, naive Bayes, and Random Forest regression (RF), achieving an Area Under the Curve of Receiver Operating Characteristic (ROCAUC) of 0.79 [6]. However, these results came from using their top 1000 features, which makes for tedious, hand-written feature extraction. Their features included n-grams, sentiment scores, and grammatical patterns. Cao et al. (2017) published a list of their top 60 clickbait features and, using an RF model as well, achieved an ROCAUC of 0.745 and an F1 score of 0.61 [2]. These results are arguably better than Potthast et al. because the model is significantly less complex and therefore less prone to over fitting. Grigorev (2017) presented a highperformance model using an ensemble of linear Support Vector Machine (SVMs) [4]. Multiple SVMs were trained on post text, keywords, captions, titles, and the articles themselves where the inputs were a Bag-of-Words word vectors (a summation of one-hot vectors). Then, the trained models were stacked using an RF. This method achieved a MSE of 0.0362. It also found that the most salient feature of clickbait is the post text. This was one of the best performing models in the Clickbait Challenge. We were influenced by the use of ensemble, and we hoped to get better results using word embeddings instead of Bag-of Words vectors. We used the dataset provided by <http://www.clickbaitchallenge.org/>, which provides 19538 data points (4761 clickbait, 14777 non-clickbait) annotated by humans. Each annotator gave a score between 0 and 1, and the median of all judgments is taken to be the true value. Each data point had the articles title, content, and the text post accompanying the article online.

III. PROPOSED SYSTEM

The proposed model can be divided into three sections as shown in Fig. 3. First, we create the data corpus by collecting clickbait and non-clickbait headlines. Then these textual headlines are converted into word embeddings which finally serves as input to our deep learning models, which in our case could be a CNN. The most difference between clickbait and normal headlines is that the linguistic character of the headline, such as questioning, exaggerating, wondering. Therefore, only the headlines are taken into consideration in the Clickbait Convolutional Neural Network (CBCNN) model. Figure 1 shows the main steps of CBCNN. When training CBCNN, the headlines are pre-processed, including segmentation, stop-words filtering, and part-of-speech filtering. After that, these headlines are utilized to train the CBCNN model. There are $1 + T$ Word2Vec [13,14] models and a CNN model present in CBCNN, where T is that the collection containing all article types, and T is that the size of T . One in all the Word2Vec models is that the overall Word2Vec model and also the other T Word2Vec models are type-related models. These Word2Vec models are utilized to embed words into the CNN model. On the predicting side, the input headline is pre-processed as the same as that of training. And then, the headline to a weight matrix is converted by using Word2Vec model. Therefore, both general characteristic and type-related characteristic are there in weight matrix. After that clickbait prediction CNN model is ready.

A. Data Collection

. Since there is an unavailability of any corpus related to clickbait, we create a corpus ourselves. Unlike others who only utilized data from one single source, we collected data from three sources, viz., Reddit, Facebook and Twitter, all three of them being popular social media platforms. We utilized this approach to ensure that the features learnt by our deep learning model don't seem to be social media platform-dependent. Each social media platform has its own limitation, eg. a maximum of 140 characters per tweet are allowed by Twitter. Therefore multiple sources of data is used to train our deep learning model for clickbait detection. To have good samples in both clickbait and non-clickbait categories, we utilized sub re-edits, pages and Fig. 4. The Proposed Model twitter handles who

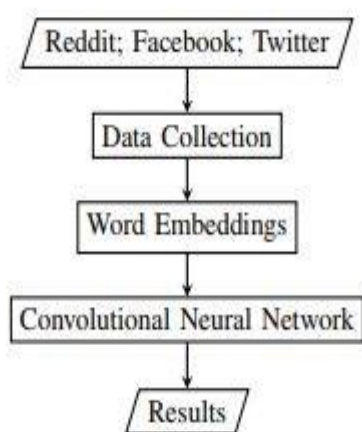


Fig. 4. The Proposed Model

have a history of publishing clickbait or non-clickbait headlines. For collecting non-clickbait headlines, we utilized the Reddit sub re-edits */r/news3* and */r/worldnews4*. These sub reedits are heavily moderated and do not allow any kind of clickbait, spam or ad to creep in. For collecting clickbait headlines, we utilized the Reddit sub re-edit */r/SavedYouAClick5* which posts only clickbait headlines with the aim to educate people and make them avoid such links. We also used Twitter handle *@HuffPoSpoilers* and the Facebook page *Stop Clickbait 6*, both of which have the same motive as */r/SavedYouAClick*. However, to maintain correctness of the data, data collected in both classes was assessed by three independent assessors. For both classes of headlines, we achieved an almost perfect inter-assessor agreement, with the clickbait headlines having a Fleiss of 0.85 while the non-clickbait headlines having a Fleiss score of 0.83. The reason for such a high value of inter-assessor agreement is due to the fact that these headlines are already in their well defined categories owing to the nature of data collection. We collected a total of 814 clickbait samples and 1574 non-clickbait samples, which have been made available online 7, by taking majority vote as the ground truth. While getting the samples, only the headlines were taken into consideration as these are the ones which create a hook in the reader or viewer. We did not take into consideration the actual web pages behind the headlines, as the headlines are the ones which first come to the attention of the reader and lure the readers to click them.

B. Word Embedding

As described in the first section, article type is a necessary feature in detecting clickbait. Different types of articles tend to write headlines in different ways, and detecting clickbait should vary accordingly. We also realize that clickbait articles have their own characteristics which should also be taken into consideration. To address this issue, we designed a new word-embedding structure for the CBCNN model. The word-embedding model consists of $1 + T$ Word2Vec models. The first one is the overall Word2Vec model, which learns from all the headlines in the data-set. The other T models are type-related Word2Vec models.

The type-related Word2Vec models learn word vectors from the headlines of the corresponding article type. The overall Word2Vec model is employed to be general clickbait characteristics, and therefore the type-related Word2Vec models learn type-related characteristics from the headlines. After that, a headline of article type t with n words is represented by $V1:n$.

$V1:n = v_0(w_1) v_0(w_2) \dots v_0(w_n)$, (1) $v_0(w) = v(w) + vt(w)$, (2) where $V1:n$ is a $k \times n$ matrix, k is the length of the word vector, is that the concatenation operator, $w_1, w_2 \dots w_n$ and w are words, t is the article type, $v_0(w)$ is w 's final word vector, $v(w)$ is w 's word vector supported the entire data-set and $vt(w)$ is w 's word vector based on articles with type t . Therefore, the word-embedding layer of the CBCNN model contains both overall characteristics and type-related characteristics, and the convolutional layer can learn more diversified features from this embedding layer. Obviously, overall characteristics are much more important than type-related characteristics. However,

they are treated equivalently in function (2). Thus within the CBCNN model, we train Word2Vec and CNN together, and add a regulation to the loss function of CBCNN. To ensure that the overall characteristics are not completely weighted of the type-related characteristics is the job of the regulation function. The new loss function is as follows: $L(X) = 1/n \sum_{x \in X} \ln[o(x)] + \lambda \sum_{t \in T} \sum_{w \in V} \frac{1}{k} \|w_j\|_2^2$, (3) where X is the training dataset, $o(x)$ is that the probability of the CBCNN models correct prediction, k is the length of the word vector, V is the vocabulary, T is that the article type, is that the weight of regularization and λ is that the size of corresponding collection. In our model, each headline x is represented by the words (w_1, w_2, \dots, w_n) appearing in it. The first part of the loss function is that of the traditional CNN model. This loss function is employed to form to make sure the prediction results of CBCNN are the same as the training data. For classification task Janocha and Czarnecki [35] suggested log loss as the best choice. Therefore, we use log loss in CBCNN. The regulation part, the second part of loss function is the one that limits type-related word vectors existing in 2-norm form. $\|w_j\|_2$ is the 2-norm of word w_j . In CBCNN, we've T types, where each type contains V words and each word consists of k dimensions, and therefore the final regulation is that the average value of all dimensions.

C. Deep Learning Models

Convolved Neural Networks (CNN) are utilized for various deep learning tasks. Here, in our present work, we use a simple CNN having one layer of convolution. The CNN we utilize is based on the CNN architecture of Kim[21]. The first layer of the CNN is used for embedding the words into vectors of low-dimensions. For word embeddings we utilize two variants word embeddings which are learnt from scratch, and word embeddings which are learnt from an unsupervised neural language model which keep evolving as training occurs. This technique of initializing word vectors from an unsupervised neural language model has been shown to improve performance [22] [23]. We utilize the word vectors trained by Mikolov, Chen, Corrado and Dean [24] on 100billion words of Google News. These vectors are publicly available as word2vec. In the next layer, filters of multiples sizes (3,

We utilize Adam [25], which is a method for stochastic optimization, for optimization of the loss function of our network. Classification is done according to the confidence index resulting from this comparison and it is decided whether or not the incoming data is Clickbait. If the confidence index is higher than 0.08 on one side and lower than 0.02 on the other, the classification of that document is finished. Values 0.08 and 0.02 were chosen to have the same distance to 0.05. Also, these are distant values to each other. Through these values, a absolute distinction can be made between the classes. If the accuracy index is higher than 0.08 in one side and lower than 0.02 on the other, the Clickbait detection process is completed for that document. In other words, if the news headlines or the subheadings frequency consistency with the Clickbait wordlist is higher than 0.08 and lower than 0.02 compared to the non-Clickbait word list, this headline is a Clickbait headline. Conversely, if the frequency match is less than 0.02 with the specified Clickbait word list and the frequency match with a non-Clickbait word list is higher than 0.08, the incoming data is not Clickbait.

IV. CONCLUSION

In this work, we will be exploring the use of variational auto-encoders for tracking the clickbait problems on Youtube, News sites and various other websites. Our approach constitutes the first proposed semi-supervised deep learning technique in the field of clickbait detection. This way, it enables more effective automated detection of clickbait videos in the absence of large-scale labeled data. We also propose using CNN framework for click-bait detection. Empirical experiments show that the model can better capture the local and global syntactic and semantic relations between words. The nuisance of clickbait keeps on increasing in online media. To curb that, we will be collecting data from multiple source

TABLE I: Accuracy and ROC-AUC scores

Model	Accuracy	ROC-AUC
Click-Scratch	0.89	0.87
Click-Word2vec	0.90	0.90

4, 5) are utilized to create convolutions over word vectors. Each such operation produces a new feature. All the new features thus generated are put into a feature map. Then, a max-over-time pooling operation is applied over the feature map and the feature with the highest value is taken as the feature for that particular feature map. The next layer, which is the penultimate layer, is formed by such generated features from the filters. These features are then passed to a fully connected soft-max layer, having the probability distribution over labels as output. For measuring loss, we use cross-entropy loss which is a standard for such categorization problems. Our aim is to minimize this loss, which represents the error in our networks.

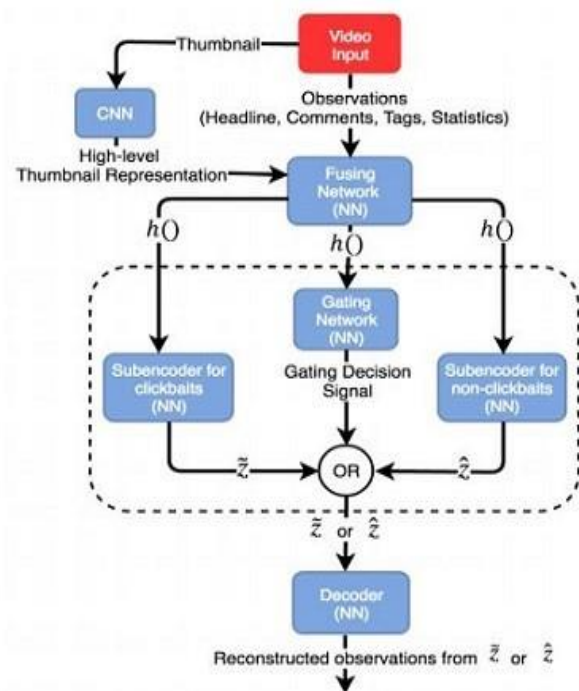


Fig. 5. Detecting Flow Diagram For Video Section

sand try to create a new corpus for clickbait and nonclickbait headlines. We then will be developing a deep learning mode;based on CNN that performs strongly on classification of headlines into clickbait and non-clickbait categories. We will be showing that these models achieve significant improvement over the state-of-the-art in detecting clickbaits without relying on heavy feature engineering. In future, we would like to qualitatively visualize the internal states of our model and incorporate attention mechanism into our model. In this paper, we proposed using CNN framework for click-bait detection. The model worked well across languages without relying on any language-specific features. We annotated a click-bait corpus of headlines collected from Chinese news feeds, which is also valuable for studying the phenomenon of click-bait in different culture.

REFERENCES

- 1) Chakraborty, A.; Paranjape, B.; Kakarla, S.; Ganguly, N. Stop clickbait: Detecting and preventing clickbait in online news media. In Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, San Francisco, CA, USA, 1821 August 2016; pp. 916.
- 2) Abbasi, A.; Chen, H. A comparison of fraud cues and classification methods for fake escrow website detection. *Inf. Technol. Manag.* 2009, 10, 83101.
- 3) Zeng, D.; Liu, K.; Chen, Y.; Zhao, J. Distant Supervision for Relation Extraction via Piecewise Convolutional Neural Networks. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Edinburgh, UK, 2731 July 2015; pp. 17531762.
- 4) Agrawal, A. Clickbait detection using deep learning. In Proceedings of the IEEE 2016 2nd International Conference on Next Generation Computing Technologies (NGCT), Dehradun, India, 1416 October 2016; pp. 268272.
- 5) Molek-Kozakowska, K. Coercive Metaphors in News Headlines a Cognitive-Pragmatic Approach. *Brno Stud. Engl.* 2014, 40, 149173.
- 6) Janocha, K.; Czarnecki, W.M. On Loss Functions for Deep Neural Networks in Classification. *Schedae Inform.* 2016, 25, 4959.
- 7) J. C. dos Reis, F. Benevenuto, P. O. S. V. de Melo, R. O. Prates, H. Kwak, and J. An, Breaking the news: First impressions matter on online news, in Proceedings of ICWSM 2015.
- 8) G. J. Digirolamo and D. L. Hintzman, First impressions are lasting impressions: A primacy effect in memory for repetitions, *Psychonomic Bulletin Review*, vol. 4, no. 1, pp. 121124, 1997. [Online]. Available: <http://dx.doi.org/10.3758/BF03210784>
- 9) D. J. Dooling and R. Lachman, Effects of comprehension on retention of prose. *Journal of Experimental Psychology*, vol. 88, no. 2, pp. 216222, 1971.
- 10) G. Loewenstein, The psychology of curiosity: A review and reinterpretation, *Psychological Bulletin*, vol. 116, no. 1, pp. 7598, July 1994
- 11) B. Vijgen, The listicle: An exploring research on an interesting shareable new media phenomenon, *Studia Universitatis Babes-Bolyai Ephemeres*, vol. 59, no. 1, pp. 103122, June 2014.
- 12) J. Nygaard Blom and K. Hansen, Click bait: Forwardreference as lure in online news headlines, *Journal of pragmatics : an interdisciplinary journal of language studies*, vol. 74, pp. 87100, 2015.
- 13) A. A. Mngen and M. Kaya, A Novel Method for Event Recommendation in Meetup, in 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2017.
- 14) T. Berners-Lee, J. Hendler, and O. Lassila, The Semantic Web, *Sci. Am.*, vol. 284, no. 5, pp. 3443, 2001.
- 15) I. Toma, D. Roman, K. Iqbal, D. Fensel, S. Decker, and J. Hofer, Towards SemanticWeb Services in Grid Environments, in *Semantics, Knowledge and Grid*, 2005. SKG 05. First International Conference on, 2005.
- 16) Y. Chen, N. J. Conroy, and V. L. Rubin, Misleading Online Content: Recognizing Clickbait as False News, in *ACM MDD*, 2015.
- 17) M. M. U. Rony, N. Hassan, and M. Yousuf, Diving Deep into Clickbaits: Who Use Them to What Extents in Which Topics with What Effects? *arXiv:1703.09400*, 2017
- 18) M. Potthast, S. Kpsel, B. Stein, and M. Hagen, Clickbait Detection, in *ECIR*, 2016.
- 19) P. Biyani, K. Tsioutsoulis, and J. Blackmer, "8 Amazing Secrets for Getting More Clicks": Detecting Clickbaits in News Streams Using Article Informality, 2016.
- 20) A. Anand, T. Chakraborty, and N. Park, We used Neural Networks to Detect Clickbaits: You wont believe what happened Next! *arXiv preprint arXiv:1612.01340*, 2016.
- 21) Y. Kim, Convolutional neural networks for sentence classification, in *Proceedings of EMNLP 2014*.
- 22) R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. P. Kuksa, Natural language processing (almost) from scratch, *Journal of Machine Learning Research*, vol. 12, pp. 24932597, 2011.
- 23) R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, Semi-supervised recursive autoencoders for predicting sentiment distributions, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP 11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 151161. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2145432.2145450>
- 24) T. Mikolov, K. Chen, G. Corrado, and J. Dean, Efficient estimation of word representations in vector space, in *Proceedings of ICLR 2013*.
- 25) D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: <http://arxiv.org/abs/1412.6980>