Clickbait Identification using Neural Networks

The Whitebait Clickbait Detector at the Clickbait Challenge 2017

Philippe Thomas
Deutsches Forschungszentrum für Künstliche Intelligenz, Germany

arXiv:1710.08721v1 [cs.CL] 24 Oct 2017

ABSTRACT

This paper presents the results of our participation in the Clickbait Detection Challenge 2017. The system relies on a fusion of neural networks, incorporating different types of available informations. It does not require any linguistic preprocessing, and hence generalizes more easily to new domains and languages. The final combined model achieves a mean squared error of 0.0428, an accuracy of 0.826, and a F_1 score of 0.564. According to the official evaluation metric the system ranked 6th of the 13 participating teams.

1. INTRODUCTION

Clickbait refers to headlines of web content targeting the human "curiosity gap" [13]. The reader is typically lured into clicking a target-link by raising interest into the advertised story mentioned in the teaser message, without providing enough details to satisfy the readers curiosity. Such clickbait-links often contain videos, picture galleries, or simple listings. The content is mostly of little journalistic quality, but spreads well in social media by referring to soft topics. Content describing such content (*e.g.,* gossip, food news, or sensational stories) is often observed in tabloid newspapers. The conversion of a newspaper into tabloid format (also referred to as tabloidization) is often considered problematic [19]. However, there are also online magazines that provide clickbait titles on more serious topics. According to an analysis all of the top 20 most prolific English news publishers on Twitter occasionally publish clickbait headlines [16]. Depending on the newspaper, percentages of clickbait content ranges from 8 % to an astonishing 51 %

In this publication we describe our approach in the Clickbait Detection Challenge 2017 [17] to detect clickbait headlines using neural networks.

2. RELATED WORK

Automated clickbait detection is a relatively recent task. Chen et al. [3] surveys potential methods and relevant concepts for the automatic detection of clickbait, including the existence of certain linguistic patterns to express clickbait headlines.

Blom and Hansen [1] hypothesized that journalists use forward-referring headlines to increase click-rates. They analyzed 100,000 headlines from 10 different Danish news for forward-reference and observed that tabloidization seem to lead to a recurrent use of forward-reference.

Chakraborty et al. [2] analyze the social sharing patterns of clickbait and non-clickbait tweets to determine the organic reach of such tweets. To this end, the authors collected a number of twitter messages from newspaper accounts known to publish a high ratio of clickbait and non-clickbait content. The authors than examine differences between these two sets in terms of consumer demographics, follower graph structure, and type of text content.

Potthast et al. [16] collected a clickbait corpus by sampling 150 tweets from each of the top 20 most prolific publishers on Twitter, totaling in 2992 tweets. This contains several renowned newspapers, as well as publishers frequently associated with clickbaiting (such as BuzzFeed or Huffington Post). All messages were rated being clickbait or not using the tweet text and the attached image. The authors also implement a clickbait detection model based on 215 features. This algorithm has been used as a baseline in the shared task. This dataset has been later extended by using crowdsourcing [18].

To the best of our knowledge previous works handled clickbait detection as binary classification task. In contrast, the organizers of the Clickbait Detection Challenge 2017 proposed a regression problem, where the task is to judge the level of clickbaiting for a given tweet. Every tweet was annotated by five individual annotators into one of the four different classes: not click baiting (0.0), slightly click baiting (0.33), considerably click baiting (0.66), or heavily click baiting (1.0). The annotations were provided as individual labels as well as different aggregation variables (*i.e.,* mean, median, mode, and class). The goal of the clickbait challenge was to accurately predict the mean value.

3. APPROACH

We used the Clickbait 2017 shared task data consisting of two labeled datasets, as described in Table 1. For internal development purpose we used the larger dataset for training and the smaller dataset for evaluation. The distribution of clickbait scores on these two datasets is shown in Figure 1 and indicates that there is a slightly higher proportion of clickbait articles in the eval-dataset. The difference between the two datasets is significant in terms of a Mann-Whitney-U test. The official test dataset is hidden to the participants but the number of true labels was revealed after the competition. Additionally, the organizers provided 80,012 unlabeled posts, which were not used in our approach.

Dataset	clickbait	no-clickbait
Train	4,761	14,777
Eval	762	1,697
Test	4,515	14,464

Table 1: Statistics of the labeled datasets. Test data was not available to participants.

Instances are provided as JSON objects and for each post we get a series of information. For the *post-tweet* this encompasses

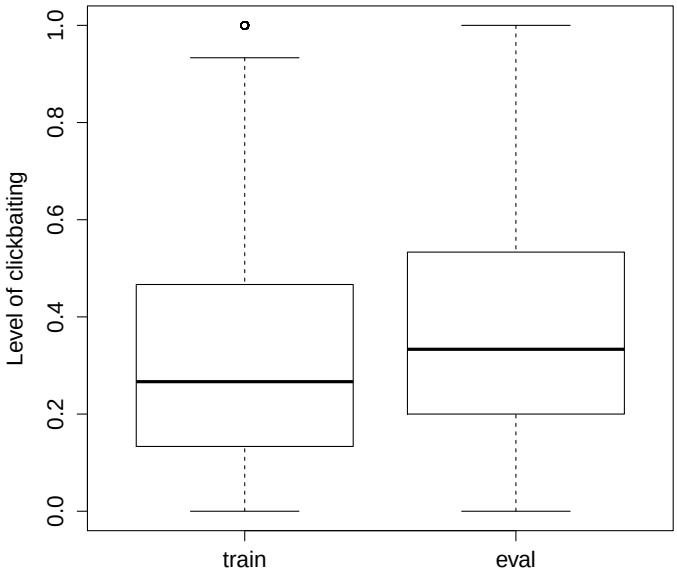


Figure 1: Distribution of clickbait scores on the two labeled datasets.

the text, potentially attached images, and the publication time. For the *target-article* this encompasses title, description, keywords, and paragraphs.

The clickbait detection approach follows closely the model used for geolocation prediction, described in Thomas and Hennig [22]. For text preprocessing, we use a simple whitespace tokenizer with lower casing, without any domain specific processing, such as unicode normalization [4] or any lexical text normalization (see for instance [8]). The texts (post-text, target-title, target-paragraphs, target-description) are converted to word embeddings [14], which are then forwarded to a Long Short-Term Memory (LSTM) unit [10]. In our experiments we randomly initialized embedding vectors. We use batch normalization [11] for normalizing inputs in order to reduce internal covariate shift. The risk of overfitting by co-adapting units is reduced by implementing dropout [21] between individual neural network layers. An example architecture for textual data is shown in Figure 2a. Post publication time (post-time) is binned into hour ranges and than converted to one-hot encodings, which are forwarded to an internal embedding layer, as proposed by Guo and Berkhahn [7]. Again batch normalization and dropout is applied to avoid overfitting. The architecture is shown in Figure 2b. We used RMSProp optimizer with early stopping and adaptive learning rate to train the neural networks. For all parameters we did not perform any systematic optimization and used 100 embedding dimensions, batch size of 32, 100 training epochs, and a dropout rate of 0.3.

Finally, individual networks are fused by concatenating the dense output layers of the individual networks. This concatenation is forwarded to a fully connected layer and used in our final model (see Figure 3 for the architecture). Similar to [22], we observed that the fusion of pre-trained models is beneficial in comparison to training a complete model from scratch.

4. EVALUATION RESULTS

Software has been uploaded to the TIRA experimentation platform [15] for automatic evaluation of all teams participating in the shared task. The TIRA environment provides a uniform environment for participants to deploy and test submissions. Using TIRA, test data is not directly available to participants and therefore avoids the possibility of information leakage.

Results of different models on our evaluation corpus are shown in Table 2. According to our analysis, post-text is, in terms of mean squared error, the most productive information resource. Using the fusion of individual neural networks we can successfully reduce the average error by 18 % (0.055 to 0.045) over the best individual model.

Model	MSE	MAS	ACC	F_1
post-text	0.055	0.153	**0.74**	**0.50**
post-time	0.057	0.19	0.69	—
target-paragraphs	0.065	0.175	0.70	0.35
target-title	0.066	0.168	0.70	0.41
target-description	0.072	0.179	0.66	0.29
target-keywords	0.073	0.194	0.68	0.20
full model	**0.045**	**0.145**	**0.74**	0.39

Table 2: Performance ranked by mean squared error (MSE) on our evaluation corpus. Other metrics include mean absolute error (MAS), accuracy (ACC) and F_1-measure.

For the final submission we combined the two labeled datasets and retrained the neural networks using the same training regime. The Whitebait clickbait detector achieved a mean squared error of 0.0428 and ranked 6th of the 13 participating teams. The official results on the test corpus (0.0428) resembles closely the error rates observed on our internal evaluation corpus (0.045).

4.1 Incorporation of image information

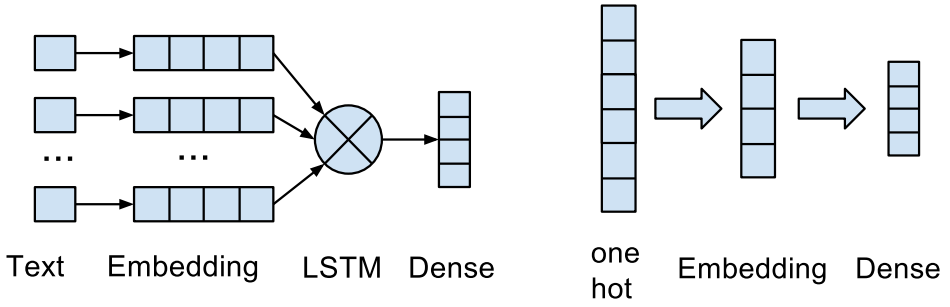
As the annotation process was supported by the image information, we assume that the teaser images might be helpful to predict the clickbait relevance of a given message. Also Ecker et al. [6] state that images can be used to attract reader attention and are usually processed before the full article is read. Therefore, we tried to incorporate the provided image information in the model. In a first account we trained a small (3 layers) convolutional neural network from scratch. Original sample size is increased using data augmentation, which modifies the original image using geometric and color augmentations [12]. In our preliminary evaluations this model achieved random performance. In future experiments we would like to apply transfer learning from deep image classification models (*e.g.,* VGG19 [20] or ResNet-50 [9]), trained on on the Image-Net dataset [5], to clickbait detection. These pre-initialized models should have already learned features, which might be relevant for our domain and should be less prone to overfitting to our data.

5. CONCLUSION

In this work we described our approach for the Clickbait Detection Challenge 2017. We implemented a neural network, relying on different information resources. The final combined model achieves a mean squared error of 0.0428, an accuracy of 0.826, and a F_1 score of 0.564. In future work we would like to incorporate the image information into the final model.

Acknowledgments

This research was partially supported by the German Federal Ministry of Economics and Energy (BMWi) through the projects SD4M (01MD15007B) and SDW (01MD15010A) and by the German Federal Ministry of Education and Research (BMBF) through the projects ALL SIDES (01IW14002) and BBDC (01IS14013E).



(a) Example architecture used for textual data. Tokenized text is represented as word embeddings, which are then forwarded to a LSTM. Dropout and batch normalization is applied between individual layers.

(b) Example architecture used for categorical data. Categorical data is represented as one-hot encodings and internally converted to entity embeddings.

Figure 2: Architectures for clickbait detection.

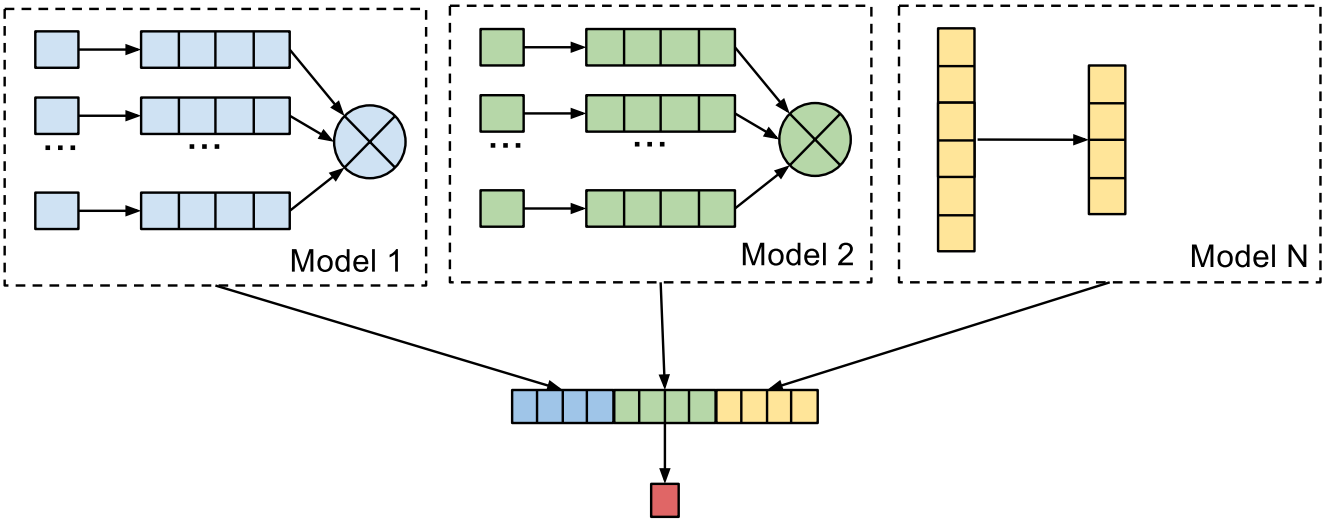


Figure 3: Neural network architecture using individual information sources.

References

[1] J. N. Blom and K. R. Hansen. Click bait: Forward-reference as lure in online news headlines. *Journal of Pragmatics*, 76 (Supplement C):87 – 100, 2015. ISSN 0378-2166.

[2] A. Chakraborty, R. Sarkar, A. Mrigen, and N. Ganguly. Tabloids in the Era of Social Media? Understanding the Production and Consumption of Clickbaits in Twitter. *ArXiv e-prints*, Sept. 2017.

[3] 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*, WMDD '15, pages 15–19, New York, NY, USA, 2015. ACM. .

[4] M. Davis, K. Whistler, and M. Dürst. Unicode Normalization Forms. Technical report, Unicode Consortium, 2001.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[6] U. Ecker, S. Lewandowsky, E. P. Chang, and R. Pillai. The effects of subtle misinformation in news headlines. *Journal of Experimental Psychology: Applied*, pages 323–335, 08 2014.

[7] C. Guo and F. Berkhahn. Entity embeddings of categorical variables. *CoRR*, abs/1604.06737, 2016.

[8] B. Han and T. Baldwin. Lexical Normalisation of Short Text Messages: Makn Sens a #Twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 368–378, Stroudsburg, PA, USA, 2011.

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[10] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780, Nov. 1997. ISSN 0899-7667. .

[11] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *CoRR*, abs/1502.03167, 2015.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[13] G. Loewenstein. The psychology of curiosity: A review and reinterpretation. *Psychological Bulletin*, pages 75–98, 1994.

[14] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. *CoRR*, abs/1310.4546, 2013.

[15] M. Potthast, T. Gollub, F. Rangel, P. Rosso, E. Stamatatos, and B. Stein. Improving the Reproducibility of PAN's Shared Tasks: Plagiarism Detection, Author Identification, and Author Profiling. In *CLEF*, pages 268–299. Springer, 2014.

[16] M. Potthast, S. Köpsel, B. Stein, and M. Hagen. Clickbait Detection. In N. Ferro, F. Crestani, M.-F. Moens, J. Mothe, F. Silvestri, G. Di Nunzio, C. Hauff, and G. Silvello, editors, *Advances in Information Retrieval. 38th European Conference on IR Research (ECIR 16)*, volume 9626 of *Lecture Notes in Computer Science*, pages 810–817, Berlin Heidelberg New York, Mar. 2016. Springer. .

[17] M. Potthast, T. Gollub, M. Hagen, and B. Stein. The Clickbait Challenge 2017: Towards a Regression Model for Clickbait Strength. In *Proceddings of the Clickbait Chhallenge*, 2017.

[18] M. Potthast, T. Gollub, K. Komlossy, S. Schuster, M. Wiegmann, E. Garces, M. Hagen, and B. Stein. Crowdsourcing a Large Corpus of Clickbait on Twitter. In *(to appear)*, 2017.

[19] D. Rowe. Obituary for the newspaper? tracking the tabloid. *Journalism*, 12(4):449–466, 2011.

[20] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.*, 15(1): 1929–1958, Jan. 2014. ISSN 1532-4435.

[22] P. Thomas and L. Hennig. Twitter geolocation prediction using neural networks. In *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology*, 9 2017.