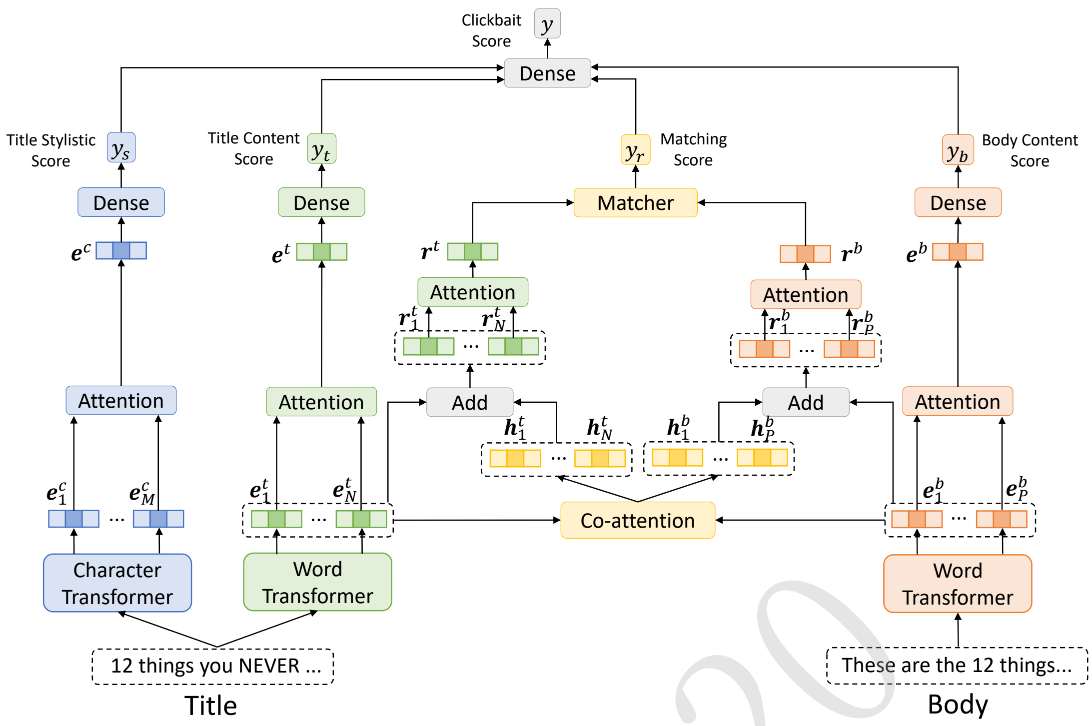



Figure 2: The architecture of our *SATC* approach for clickbait detection.

## 3.1 Content Modeling

The *content modeling* module is used to learn the representations of title and body from their content. We respectively denote the sequences of words in title and body as $[w_1^t, w_2^t, ..., w_N^t]$ and $[w_1^b, w_2^b, ..., w_P^b]$, where $N$ and $P$ respectively stand for the number of words in the title and body. In this module, we first use a word embedding layer to convert both word sequences into sequences of semantic vectors, which are denoted as $[\mathbf{w}_1^t, \mathbf{w}_2^t, ..., \mathbf{w}_N^t]$ and $[\mathbf{w}_1^b, \mathbf{w}_2^b, ..., \mathbf{w}_P^b]$. Usually the contexts of words in title and body are important for modeling their content. For example, in the title of the first webpage in Fig. 1, the contexts of the word "Loss" such as "Weight" and "Exercise" are useful clues for understanding that this word is about fitness rather than financial loss. Transformer (Vaswani et al., 2017) is an effective neural architecture for context modeling. Thus, we apply two independent Transformers to learn hidden representations of words in title and body by modeling their contexts. We denote the hidden representation sequences of words in title and body as $\mathbf{E}^t = [\mathbf{e}_1^t, \mathbf{e}_2^t, ..., \mathbf{e}_N^t]$ and $\mathbf{E}^b = [\mathbf{e}_1^b, \mathbf{e}_2^b, ..., \mathbf{e}_P^b]$, respectively. Different words in a title or body may have different importance for modeling the content. For instance, the word "MUST" in Fig. 1 is more important than the word "About" in learning title representation for clickbait detection. Thus, we apply attention mechanisms (Yang et al., 2016) to select words in the title and body to form unified representations for them (denoted as $\mathbf{e}^t$ and $\mathbf{e}^b$), which are respectively formulated as follows:

$$\mathbf{e}^t = Attention([\mathbf{e}_1^t, \mathbf{e}_2^t, ..., \mathbf{e}_N^t]), \tag{1}$$

$$\mathbf{e}^b = Attention([\mathbf{e}_1^b, \mathbf{e}_2^b, ..., \mathbf{e}_P^b]). \tag{2}$$

## 3.2 Style Modeling

The *style modeling* module is used to capture the stylistic patterns in the title to better identify clickbaits. Usually, there are some common patterns on the style of clickbait titles. For example, many clickbaits use all-capital words (e.g., "MUST", "NOT" and "THIS"), exclamation marks, and numeric characters to attract users' attention. Thus, it is very important to grasp these stylistic patterns in clickbait detection. To capture these patterns, we propose to use a character-level Transformer to learn style-aware title

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154, Hainan, China, October 31 - Novermber 1, 2020.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

representations from its original characters. We denote the character sequence (including whitespace) of the title as $[c_1, c_2, ..., c_M]$, where $M$ is the number of characters. We first convert these characters into their embeddings (denoted as $[\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_M]$) via a character embedding layer, and then use a character Transformer to learn the hidden representations of these characters, which are denoted as $[\mathbf{e}_1^c, \mathbf{e}_2^c, ..., \mathbf{e}_M^c]$. Usually different characters may have different importance in style modeling. For example, in Fig. 1 the character "7" is more important than the character "a" in the word "and". Thus, we use a character-level attention network for character selection in building the style-aware title representation $\mathbf{e}^c$, which is formulated as follows:

$$\mathbf{e}^c = Attention([\mathbf{e}_1^c, \mathbf{e}_2^c, ..., \mathbf{e}_M^c]). \tag{3}$$

### 3.3 Interaction Modeling

The *interaction modeling* module is used to capture the interactions between title and body. For most webpages, the contexts in their titles usually have relatedness with the contexts in their bodies to a certain extent. For instance, the words "Restaurants" in the title of the third webpage in Fig. 1 have close relatedness with the words "businesses", "restaurants" and "cafes" in the body. These interactions are important cues for modeling the relevance between title and body, which is critical for clickbait detection. Thus, we propose to use a multi-head co-attention network to capture the interactions between title and body. More specifically, we first use the title word representation sequence $\mathbf{E}^t$ as the query, and use the body word representation sequence $\mathbf{E}^b$ as the key and value to compute a hidden representation sequence $\mathbf{H}^t = [\mathbf{h}_1^t, \mathbf{h}_2^t, ..., \mathbf{h}_N^t]$, which summarizes the contexts within body and their interactions with each word in the title. This process is formulated as follows:

$$\mathbf{H}^t = MultiHead(\mathbf{E}^t, \mathbf{E}^b, \mathbf{E}^b). \tag{4}$$

Next, we use the body word representation sequence $\mathbf{E}^b$ as the query, and use the title word representation sequence $\mathbf{E}^t$ as the key and value to compute an hidden representation sequence $\mathbf{H}^b = [\mathbf{h}_1^b, \mathbf{h}_2^b, ..., \mathbf{h}_P^b]$ that conveys the contexts in title and their interactions with each word in body, which is formulated as follows:

$$\mathbf{H}^b = MultiHead(\mathbf{E}^b, \mathbf{E}^t, \mathbf{E}^t). \tag{5}$$

Then, we use the interactions between title and body to enhance their representations. We add the hidden representation sequence $\mathbf{H}^t$ to the original word representation sequence $\mathbf{E}^t$ to form a unified representation sequence $\mathbf{R}^t$, i.e., $\mathbf{R}^t = \mathbf{E}^t + \mathbf{H}^t$. The unified body word representation sequence $\mathbf{R}^b$ is obtained by $\mathbf{R}^b = \mathbf{E}^b + \mathbf{H}^b$. Similar to the *content modeling* module, we also use attention networks to obtain the final interaction-enhanced representations of title and body (denoted as $\mathbf{r}^t$ and $\mathbf{r}^b$), which are formulated as follows:

$$\mathbf{r}^t = Attention([\mathbf{r}_1^t, \mathbf{r}_2^t, ..., \mathbf{r}_N^t]), \tag{6}$$

$$\mathbf{r}^b = Attention([\mathbf{r}_1^b, \mathbf{r}_2^b, ..., \mathbf{r}_P^b]), \tag{7}$$

where $\mathbf{r}_i^t$ and $\mathbf{r}_i^b$ stand for the $i$-th vector in $\mathbf{R}^t$ and $\mathbf{R}^b$, respectively.

### 3.4 Clickbait Prediction

The *clickbait prediction* module is used to compute a clicbait score based on the representations of title and body. We first use a dense layer to compute a title content score $y_t$ based on the content representation $\mathbf{e}^t$ of the title, which is formulated as $y_t = \mathbf{w}_t^\top \mathbf{e}^t + b_t$, where $\mathbf{w}_t$ and $b_t$ are the kernel and bias parameters. We compute a body content score $y_b$ based on $\mathbf{e}^b$ in a similar way, which is formulated as $y_b = \mathbf{w}_b^\top \mathbf{e}^b + b_b$, where $\mathbf{w}_b$ and $b_b$ are parameters. Next, we use a matcher to compute a title-body matching score, which indicates the relevance between title and body. It takes the interaction-enhanced representations of title and body ($\mathbf{r}^t$ and $\mathbf{r}^b$) as the input, and outputs the matching score $y_r$. Following (Okura et al., 2017), we use dot-product to implement the matcher, and the score $y_r$ is computed as $y_r = \mathbf{r}^t \cdot \mathbf{r}^b$. Then, we use another dense layer to compute a title stylistic score based on the style-aware title representation $\mathbf{e}^c$, which is formulated as $y_s = \mathbf{w}_s^\top \mathbf{e}^s + b_s$, where $\mathbf{w}_s$ and $b_s$ are parameters. The final clickbait score $y$ is a

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154, Hainan, China, October 31 - November 1, 2020.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

weighted summation of the aforementioned four scores and we use the sigmoid function for normalization, which is formulated as follows:

$$y = sigmoid(\alpha_s y_s + \alpha_t y_t + \alpha_r y_r + \alpha_b y_b), \quad (8)$$

where $\alpha_s$, $\alpha_t$, $\alpha_r$ and $\alpha_b$ are trainable parameters.

For model training, we use binary cross-entropy as the loss function. By comparing the predicted clickbait score with the gold label, we can obtain the loss on the training samples, and further compute the gradients for model update.

## 4 Experiments

### 4.1 Dataset and Experimental Settings

Our experiments are conducted on two benchmark datasets for clickbait detection. The first one is *Clickbait Challenge*[1], which is a dataset released by the organizers of Clickbait Challenge 2017. This dataset contains the tweet texts posted by users and the content of the corresponding article. Each pair of tweet and article is annotated by 5 judgers, where each judger gives a clickbait score from 0 (non-clickabit) to 1 (clickbait) to this pair. Following (Dong et al., 2019), we regard the pairs with the mean score over 0.5 as clickbaits. The training set contains 19,538 pairs, and the validation set contains 2,495 pairs. Since the labels of the test set are not released, we evaluate the model on the current validation set, and randomly sample 10% of pairs in the training set for validation. The second one is *FNC*[2], which is released by the Fake News Challenge in 2017. In this dataset, each pair of title and body is labeled as “agree”, “disagree”, “discuss” or “unrelated”. Following (Dong et al., 2019), we regard the pairs with “unrelated” labels as clickbaits. This dataset contains 49,972 pairs of titles and bodies for training and 25,413 pairs for test. We also use 10% of training samples for validation.

In our experiments, we use the pre-trained 300-dimensional Glove embeddings (Pennington et al., 2014) to initialize the parameters in the word embedding layer. We do not fine-tune these pre-trained word embeddings in model training to avoid overfitting. The character embeddings are 50-dimensional. The Transformers have two self-attention layers. Each layer has 8 attention heads, and the output dimension of each head is 32. We apply dropout (Srivastava et al., 2014) to the word and character embeddings at a ratio of 20%. We use Adam (Kingma and Ba, 2014) as the optimizer, and the learning rate is 0.01. The size of each mini-batch is 64. These hyperparameters are searched according to the performance on the validation sets. Each experiment is repeated 5 times, and the average results in terms of accuracy, precision, recall and Fscore are reported.

CCL 2020

### 4.2 Performance Evaluation

We compare our *SATC* method with several baseline methods, including:

- DSSM (Huang et al., 2013), deep structured semantic model, where title is regarded as the query and body is regarded as document. The texts of title and body are represented by N-gram featuress.
- CLSM (Shen et al., 2014), a variant of DSSM that uses CNN to learn text representations;
- CNN (Agrawal, 2016; Zheng et al., 2018), which detects clickbaits solely based on titles. Text-CNN is used to learn title representations.
- LSTM (Glenski et al., 2017), using LSTM networks to learn title and body representations for clickbait detection.
- GRU-Att (Zhou, 2017), using a combination of bi-GRU network and attention network to learn title representations for clickbait detection.

[1]https://www.clickbait-challenge.org/.
[2]http://www.fakenewschallenge.org/

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154, Hainan, China, October 31 - Novermber 1, 2020.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

| Method | Clickbait Challenge | | | | FNC | | | |
|---|---|---|---|---|---|---|---|---|
| | Accuracy | Precision | Recall | Fscore | Accuracy | Precision | Recall | Fscore |
| DSSM | 0.817 | 0.655 | 0.661 | 0.658 | 0.747 | 0.894 | 0.740 | 0.811 |
| CLSM | 0.833 | 0.683 | 0.643 | 0.662 | 0.756 | 0.959 | 0.762 | 0.853 |
| CNN | 0.844 | 0.654 | 0.653 | 0.653 | 0.789 | 0.852 | 0.845 | 0.857 |
| LSTM | 0.827 | 0.642 | 0.621 | 0.631 | 0.868 | 0.925 | 0.884 | 0.913 |
| GRU-Att | 0.856 | 0.719 | 0.650 | 0.683 | 0.879 | 0.924 | 0.897 | 0.919 |
| Siamese Net | 0.844 | 0.695 | 0.688 | 0.691 | 0.859 | 0.920 | 0.877 | 0.907 |
| LSDA | 0.860 | 0.697 | 0.699 | 0.710 | 0.894 | 0.933 | 0.912 | 0.928 |
| SATC* | **0.889** | **0.745** | **0.722** | **0.733** | **0.907** | **0.959** | **0.917** | **0.938** |

Table 1: Performance comparison of different methods on the two datasets. *Improvement is significant at the level of $p < 0.01$.

- SiameseNet (Kumar et al., 2018), which uses *GRU-Att* to learn title representations and uses Siamese networks to capture the relevance between title and body.
- LSDA (Dong et al., 2019), which uses *GRU-Att* to learn title and body representations, and measures their relevance using the global and local similarities between the representation vectors of title and body.

The results on the two datasets are summarized in Table 1.[3] According to the results, we have several main findings. First, the methods that use neural networks to learn text representations (e.g., *CNN*, *LSTM*, *GRU-Att* and *SATC*) outperform the *DSSM* method that uses handcrafted features for text representation. It shows that handcrafted features are usually not-optimal in representing the textual content of webpages for clickbait detection. Second, the methods based on attention mechanisms (e.g., *GRU-Att* and *LSDA*) usually outperform the methods without attention (e.g., *CNN* and *LSTM*). This is probably because attention mechanism can select important contexts within title and body to learn more informative representations for them, which is beneficial for clickbait detection. Third, our approach can consistently outperform the compared baseline methods. This is because our approach can capture the stylistic patterns in the title to learn style-aware title representations, and meanwhile can model the interactions between contexts in title and body to help measure their relevance more accurately. In addition, Transformers may also have a greater ability than CNN, LSTM and GRU in context modeling. Thus, our method can detect clickbaits more effectively than baseline methods.

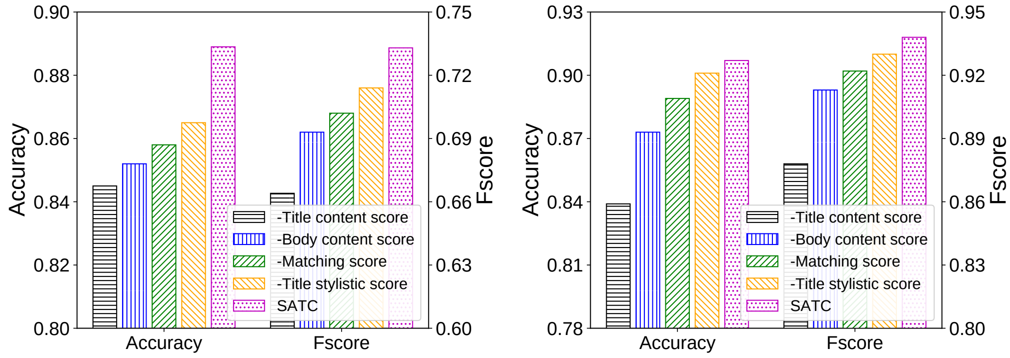
### 4.3 Influence of Different Scores

In this section, we conduct several ablation studies to explore the influence of the four clickbait scores. We compare the performance of our *SATC* approach by removing one of these scores in clickbait prediction. The results on the *Clickbait Challenge* and *FNC* datasets are respectively shown in Figs. 3(a) and 3(b). From the results, we find that the title content score plays the most important role. This is intuitive because clickbaits mainly rely on the content of their titles to attract users' attention and clicks. Thus, modeling the title content is critical for clickbait detection. In addition, we find the body content score is also important. This is because the body of many clickbaits may be misleading or uninformative. Thus, modeling the content of body is important for clickbait detection. Besides, the matching score is also useful for clickbait prediction. This is probably because the titles of some clickbaits do not perfectly match their bodies. Thus, modeling the relevance of title and body is useful for accurate clickbait detection. Moreover, we find the title stylistic score is also helpful. This is mainly because the stylistic patterns of title are important clues for identifying clickbaits, but these clues may not be captured by the content modeling module. Thus, the title stylistic score can provide complementary information to help detect clickbaits better. These results verify the effectiveness of the four different clickbait scores in our approach.

[3] Most results of baselines are taken from (Dong et al., 2019), except the result of Siamese Net on the *Clickbait Challenge* dataset since it is quite unsatisfactory. We report the results using our implementation instead.

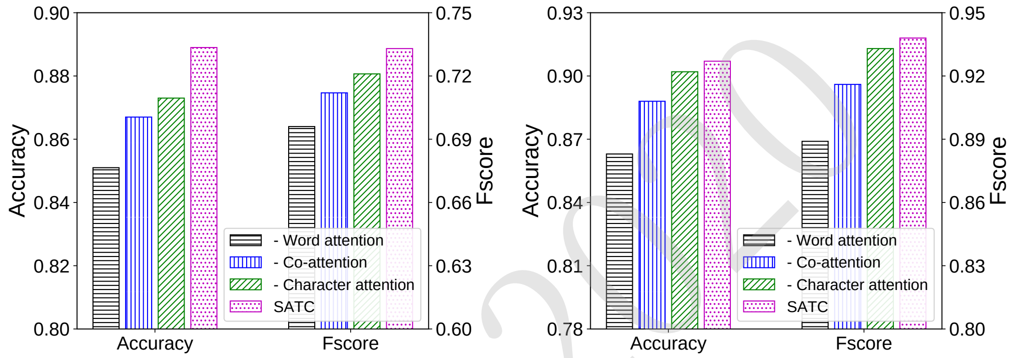
CCL 2020

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154, Hainan, China, October 31 - Novermber 1, 2020.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China




(a) *Clickbait Challenge* dataset. (b) *FNC* dataset.

Figure 3: Influence of removing different scores in clickbait prediction.




(a) *Clickbait Challenge* dataset. (b) *FNC* dataset.

Figure 4: Effectiveness of different attention networks.

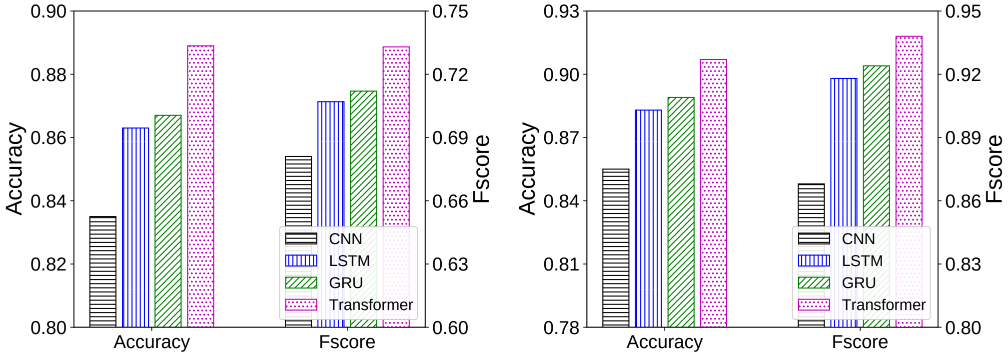
## 4.4 Effectiveness of Attention Mechanism

In this section, we verify the effectiveness of the word-level attention, character-level attention and co-attention networks in our approach. More specifically, we compare the performance of our *SATC* approach and its variants without one kind of attention. The results on the *Clickbait Challenge* and *FNC* datasets are respectively shown in Figs. 4(a) and 4(b). We find that the word-level attention network is very helpful. This may be because different words are usually diverse in their informativeness and the work-level attention networks can attend to the important words in title and body, which can help learn more informative representations of them. In addition, the co-attention network can also effectively improve the model performance. This may be because the co-attention network can model the interactions of words in title and body and can further enhance the title and body representations by encoding interaction information, which is beneficial for evaluating the relevance between title and body. Besides, the character-level attention network can also improve the performance to some extent. This may be because different characters also have different importance in modeling the stylistic patterns of the title and the character-level attention network is able to select useful characters, which can help learn more informative style-aware title representations.

## 4.5 Effectiveness of Transformer

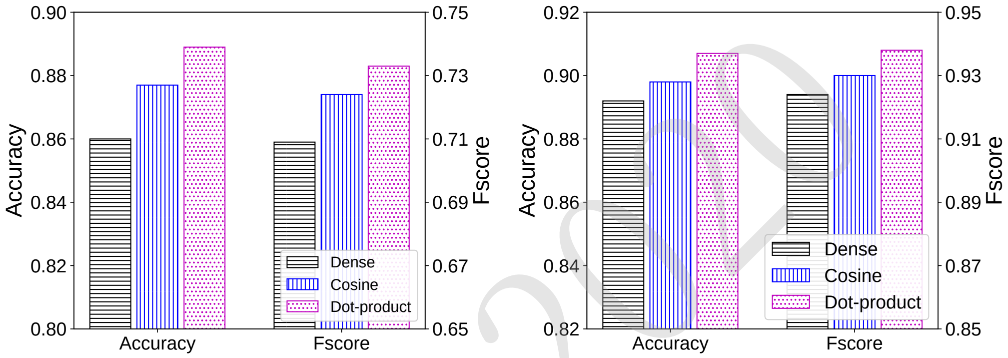
In this section, we verify the effectiveness of Transformers in text modeling in our approach. We compare the performance of *SATC* and its several variants using CNN, LSTM and GRU for text modeling, and the results are illustrated in Figs. 5(a) and 5(b). From the results, we find that using CNN is not optimal in text modeling for clickbait detection. This is because CNN can only capture local contexts, while the


(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China




(a) *Clickbait Challenge* dataset. (b) *FNC* dataset.

Figure 5: Effectiveness of Transformer in text modeling.




(a) *Clickbait Challenge* dataset. (b) *FNC* dataset.

Figure 6: Influence of using different methods for computing matching scores.

CCL 2020

long-distance contexts are not considered. In addition, we find GRU slightly outperforms LSTM. This may be because the GRU networks contain fewer parameters and have a lower risk of overfitting. Besides, Transformer outperforms LSTM and GRU. This is because Transformer is very effective in modeling the relations between contexts, which has also been validated by existing works (Vaswani et al., 2017). Thus, we prefer Transformer for learning text representations for clickbait detection.

### 4.6 Influence of Matching Methods

In this section, we explore the influence of using different methods to implement the matcher in our approach to compute the matching score. We compare the performance of *SATC* using dot-product, dense network and cosine similarity as the matcher. The results are illustrated in Figs. 6(a) and 6(b). From the results, we find that using a dense network is not optimal. According to (Rendle et al., 2020), a possible reason is that dense network is difficult to measure the similarity between two vectors, and thereby the matching score may be inaccurate. In addition, we find that using dot-product is slightly better than using cosine similarity. This may be because the cosine similarity function is not sensitive to the length of the input vectors, which may not be optimal for measuring the relevance between the title and body. Thus, we choose dot-product to implement the matcher in our method.

### 4.7 Case Study

In this section, we conduct several case studies to better understand the characteristics of our approach. The title, body, groundtruth and the predictions results of *GRU-Att*, *LSDA* and our *SATC* on several samples are shown in Table 2, and we have several findings. In Table 2, the first sample is a clickbait because its title does not match its body. However, since the *GRU-Att* method only considers the information of

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154, Hainan, China, October 31 - Novermber 1, 2020.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

| Title | Body | Label | Prediction | | |
|---|---|---|---|---|---|
| | | | GRU-Att | LSDA | SATC |
| Report: NHL expansion to Las Vegas'a done deal' | Brain surgery recovery can be a gamble, but not everybody wakes up in the middle of the procedure... | 1 | 0.07 | 0.88 | 0.95 |
| The real-life Indiana Jane will make you sooooooooooo jealous of her life | Meet the real-life Indiana Jane: American adventurer spends her life in dangerous jungles and uncharted wildernesses... | 1 | 0.23 | 0.16 | 0.98 |
| Apple Watch may be available outside US shortly after launch | Lately, Apple CEO has been making the rounds in Europe, stopping at various stores and chatting with employees. The last time we heard anything about his commentary on Apple Watch... | 0 | 0.12 | 0.68 | 0.05 |

Table 2: The titles, bodies, labels and the predicted scores of different methods on several samples. 0 stands for non-clickbait and 1 stands for clickbait.

title, it fails to detect this clickbait. The other two methods that consider the relevance between title and body classify this sample correctly. Thus, it is important to model the title-body relevance for clickbait detection. The title of the second sample in Table 2 contains a word with repeated characters to express strong emotion, which is an important indication of clickbaits. However, this word is out-of-vocabulary, making it difficult for the *GRU-Att* and *LSDA* methods to capture this clue. Thus, these methods fail to detect this clickbait. Different from them, our approach uses a character-level Transformer to capture the stylistic patterns in the title, and thereby can detect this clickbait at a high confidence. The third sample in Table 2 is not a clickbait because the title is formal and the title is relevant to the body. However, it is not easy to measure the relevance between the title and body of this sample without considering the interactions between their words, since the body does not frequently mention the words like "US" and "Watch" that appear in the title. Thus, the *LSDA* method, which does not consider the interactions between contexts, incorrectly classifies this sample as a clickbait. Since our approach uses a co-attention network to model title-body interactions, it classifies this sample correctly.

## 5 Conclusion

In this paper, we propose a clickbait detection approach with style-aware title modeling and co-attention, which can capture the stylistic patterns in the title and the interactions between the contexts in the title and body. We use Transformers to learn content representations of title and body, and respectively compute two content-based clickbait scores for them based on their representations. In addition, we propose to apply a character-level Transformer to capture the stylistic patterns of title for learning style-aware title representations, which are further used to compute a title stylistic score. Besides, we propose to use a co-attention network to model the relatedness between the contexts within title and body, and further combine their original representations with the interaction information to learn interaction-enhanced title and body representations, which are further used to compute a title-body matching score. The final clickbait score is predicted by a weighted summation of the four kinds of clickbait scores. Extensive experiments on two benchmark datasets show that our approach can effectively improve the performance of clickbait detection by using style-aware title modeling to capture stylistic information and co-attention networks to model title-body interactions.

## Acknowledgments

This work was supported by the National Key Research and Development Program of China under Grant number 2018YFC1604002, the National Natural Science Foundation of China under Grant numbers U1936208, U1936216, U1836204, and U1705261.

CCL 2020

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154, Hainan, China, October 31 - Novermber 1, 2020.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

## References


Amol Agrawal. 2016. Clickbait detection using deep learning. In *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, pages 268–272. IEEE.

Ankesh Anand, Tanmoy Chakraborty, and Noseong Park. 2017. We used neural networks to detect clickbaits: You won't believe what happened next! In *ECIR*, pages 541–547. Springer.

Prakhar Biyani, Kostas Tsioutsiouliklis, and John Blackmer. 2016. " 8 amazing secrets for getting more clicks": Detecting clickbaits in news streams using article informality. In *AAAI*.

Peter Bourgonje, Julian Moreno Schneider, and Georg Rehm. 2017. From clickbait to fake news detection: an approach based on detecting the stance of headlines to articles. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 84–89.

Xinyue Cao, Thai Le, et al. 2017. Machine learning based detection of clickbait posts in social media. *arXiv preprint arXiv:1710.01977*.

Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. 2016. Stop clickbait: Detecting and preventing clickbaits in online news media. In *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 9–16. IEEE.

Yimin Chen, Niall J Conroy, and Victoria L Rubin. 2015. Misleading online content: recognizing clickbait as" false news". In *Proceedings of the 2015 ACM on workshop on multimodal deception detection*, pages 15–19.

Philogene Kyle Dimpas, Royce Vincent Po, and Mary Jane Sabellano. 2017. Filipino and english clickbait detection using a long short term memory recurrent neural network. In *IALP*, pages 276–280. IEEE.

Manqing Dong, Lina Yao, Xianzhi Wang, Boualem Benatallah, and Chaoran Huang. 2019. Similarity-aware deep attentive model for clickbait detection. In *PAKDD*, pages 56–69. Springer.

Junfeng Fu, Liang Liang, Xin Zhou, and Jinkun Zheng. 2017. A convolutional neural network for clickbait detection. In *2017 4th International Conference on Information Science and Control Engineering (ICISCE)*, pages 6–10. IEEE.

Ayse Geçkil, Ahmet Anil Müngen, Esra Gündogan, and Mehmet Kaya. 2018. A clickbait detection method on news sites. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 932–937. IEEE.

Maria Glenski, Ellyn Ayton, Dustin Arendt, and Svitlana Volkova. 2017. Fishing for clickbaits in social images and texts with linguistically-infused neural network models. *arXiv preprint arXiv:1710.06390*.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *CIKM*, pages 2333–2338.

Vijayasaradhi Indurthi and Subba Reddy Oota. 2017. Clickbait detection using word embeddings. *arXiv preprint arXiv:1710.02861*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Vaibhav Kumar, Dhruv Khattar, Siddhartha Gairola, Yash Kumar Lal, and Vasudeva Varma. 2018. Identifying clickbait: A multi-strategy approach using neural networks. In *SIGIR*, pages 1225–1228.

Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. 2017. Embedding-based news recommendation for millions of users. In *KDD*, pages 1933–1942.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016. Clickbait detection. In *ECIR*, pages 810–817. Springer.

Steffen Rendle, Walid Krichene, Li Zhang, and John Anderson. 2020. Neural collaborative filtering vs. matrix factorization revisited. *arXiv preprint arXiv:2005.09683*.

Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A latent semantic model with convolutional-pooling structure for information retrieval. In *CIKM*, pages 101–110.


CCL 2020

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154, Hainan, China, October 31 - Novermber 1, 2020.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958.

Philippe Thomas. 2017. Clickbait identification using neural networks. *arXiv preprint arXiv:1710.08721*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *NAACL-HLT*, pages 1480–1489.

Hai-Tao Zheng, Jin-Yuan Chen, Xin Yao, Arun Kumar Sangaiah, Yong Jiang, and Cong-Zhi Zhao. 2018. Clickbait convolutional neural network. *Symmetry*, 10(5):138.

Yiwei Zhou. 2017. Clickbait detection in tweets using self-attentive network. *arXiv preprint arXiv:1710.05364*.

CCL 2020

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154, Hainan, China, October 31 - Novermber 1, 2020.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China