Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.040095
Article

Predicting the Popularity of Online News Based on the Dynamic Fusion of Multiple Features

Guohui Song[1,2], Yongbin Wang[1,*], Jianfei Li[1] and Hongbin Hu[1]

[1]State Key Laboratory of Media Convergence and Communication, Communication University of China, Beijing, 100024, China
[2]School of Computer and Cyber Sciences, Communication University of China, Beijing, 100024, China
*Corresponding Author: Yongbin Wang. Email: ybwang@cuc.edu.cn
Received: 04 March 2023; Accepted: 19 May 2023; Published: 30 August 2023

Abstract: Predicting the popularity of online news is essential for news providers and recommendation systems. Time series, content and meta-feature are important features in news popularity prediction. However, there is a lack of exploration of how to integrate them effectively into a deep learning model and how effective and valuable they are to the model's performance. This work proposes a novel deep learning model named Multiple Features Dynamic Fusion (MFDF) for news popularity prediction. For modeling time series, long short-term memory networks and attention-based convolution neural networks are used to capture long-term trends and short-term fluctuations of online news popularity. The typical convolution neural network gets headline semantic representation for modeling news headlines. In addition, a hierarchical attention network is exploited to extract news content semantic representation while using the latent Dirichlet allocation model to get the subject distribution of news as a semantic supplement. A factorization machine is employed to model the interaction relationship between meta-features. Considering the role of these features at different stages, the proposed model exploits a time-based attention fusion layer to fuse multiple features dynamically. During the training phase, this work designs a loss function based on Newton's cooling law to train the model better. Extensive experiments on the real-world dataset from Toutiao confirm the effectiveness of the dynamic fusion of multiple features and demonstrate significant performance improvements over state-of-the-art news prediction techniques.

Keywords: Attention mechanism; deep learning; time series; popularity prediction

1 Introduction

In China, news platforms such as Toutiao and Tencent News have become the primary way for users to obtain news, and the number of monthly active users has reached hundreds of millions. Predictions of news popularity are valuable to news platforms and content providers. For example,

predicting and monitoring news popularity can help improve the platform's recommendation strategy. Similarly, using the popularity prediction method, resources can be better allocated to meet the evolving information needs of readers.

The research on news popularity prediction has drawn significant attention for decades. News popularity describes how much user attention a piece of news has gained and can be measured by the number of clicks or views. The prediction can occur before or after the news article is published, and this work focuses on the latter. News popularity prediction is generally regarded as a regression or classification problem relying on text, meta-features, or time series. According to the category of input features, there are two main categories of existing modeling methods: time series modeling and content feature modeling.

Content features contain textual features and meta-features. Textural features mainly include the headline and content of the news article, while meta-features refer to the article's author, source, publishing agency, and publishing time. Some works adopt classical machine learning methods for predicting news popularity [1–7]. The performance of these models is highly dependent on the extracted features. However, the extraction and measurement of these features are complex. Due to the ability of deep learning to capture features automatically, it is widely used in news popularity prediction [8,9]. However, how to effectively represent and integrate different forms of text and metadata features has not been further studied. Time series modeling defines news popularity as the cumulative process of user browsing behavior based on time series. These works mainly applied the long short-term memory networks (LSTM) model for time series modeling [10,11]. The methods based on time series can predict better over time, but they have the drawback of relying on a series of historical user events. Therefore, getting the overall trend at the beginning of news published is not easy. As a result, news content feature modeling is more reliable in the early stages, while it fails to take advantage of the temporal evolution of popularity. Some works adopt multi-feature fusion methods to take advantage of time series and content features [12–14]. However, how to effectively represent and flexibly integrate the two types of features to exert their respective performance at different stages requires further research.

To address the above limitations of existing research, this work proposes a novel neural network model named Multiple Features Dynamic Fusion (MFDF) for news popularity prediction. The proposed model effectively represents a variety of news features and dynamically fuses them. Time series modeling captures long-term and short-term trends of news popularity by LSTM and convolutional neural networks (CNN). Content feature modeling comprises text (headline, content) semantic representations and the interaction relationship between meta-features. The typical CNN-based model [15] is used for modeling news headlines' semantic representation. For modeling news content, the hierarchical attention network (HAN) [16] is exploited to extract text features while using the latent Dirichlet allocation(LDA) [17] to get the subject distribution of news as a semantic supplement. Meanwhile, the factorization machine (FM) [18] is employed to model the interaction relationship between meta-features. Considering the role of these features at different stages, this study uses a time-based attention layer to fuse these features dynamically. Besides, this work designs a loss function based on Newton's cooling law to train the model better. Extensive experiments on the real-world dataset from Toutiao confirm the effectiveness of the dynamic fusion of multiple features and demonstrate significant performance improvements over state-of-the-art news prediction techniques.

In summary, the key contributions of this paper are summarized below:

(1) To our best knowledge, this work is the first to fuse time series, content features, and high-level meta-feature interactions in online news popularity prediction.

(2) This paper proposed a deep learning approach, MFDF, that fuses multiple features dynamically by a time-based attention layer. Besides, this work designs a loss function based on Newton's cooling law to train the model better.

(3) Extensive experiments demonstrate significant performance improvements over state-of-the-art news prediction techniques, confirming the validity of the proposed model.

The rest of this paper is organized as follows. The next Section summarizes the current work related to this study. Section 3 describes the detail of the proposed model. Section 4 presents the extensive experimental results and detailed analysis. Section 5 finally concludes this paper.

2 Related Work

According to how to extract features, the predictive models can be divided into two categories: classical machine learning methods and deep learning methods.

2.1 Classical Machine Learning Methods

Classical machine learning methods require feature engineering to select predicted features. Keneshloo et al. [1] extracted meta, content, and temporal features and adopted tree regression to predict news popularity. Choudhary et al. [2] adopted the similarity coefficient to extract 32 attributes as input of news popularity prediction models constructed by Naive Bayes and Random Forest algorithms. Khan et al. [3] extracted the first ten features as input of an integrated classifier, improving prediction accuracy. Tsai et al. [5] attempted to combine an Auto-encoder and One-Class support vector machine (SVM) algorithms to predict news popularity. Yang et al. [7] proposed a named entity topic model to extract textual factors that can promote news popularity. By learning the popularity-gain matrix for each named entity, it is possible to predict the popularity of any news article. Rajagopal et al. [19] extracted two types of features in the news: general features containing metadata, temporal, context, and embedding vector features, and augmented features extracted from articles, including readability, emotion, and psycholinguistic features. Experiments with multiple supervised learning models show the effectiveness of combining enhanced and standard features for popularity prediction. There are some issues with classic machine learning methods for news popularity prediction. First, those approaches are highly sensitive to the selected features. Second, the traditional machine learning methods usually adopt the bag-of-words model for indicating textual features of news without considering text semantics.

2.2 Deep Learning Methods

Due to the enormous success of deep learning, some researchers leverage the neural network to avoid feature engineering. Guan et al. [8] proposed a hierarchical neural network to learn text representations for news popularity prediction. Stokowiec et al. [20] obtained a textual representation of headlines using bidirectional LSTM (Bi-LSTM) to predict the popularity of news on Facebook. These works have achieved significant success, demonstrating the great advantages of deep learning architectures in predicting news popularity. Some neural network structures such as recurrent neural networks (RNN) [21], LSTM [22], Bi-LSTM [23] and gated recurrent units (GRU) [24] have been widely used in the processing of news text representation. In addition, some convolutional neural networks have also been applied to the text representation, such as CNN [15] and gated convolutional neural network (GCNN) [25]. Some advanced pre-trained models, such as bidirectional encoder representations from transformers (BERT), are gaining momentum and have successfully succeeded in natural language processing [26,27]. This study focuses mainly on designing a deep learning model for

news popularity prediction rather than a pre-trained model for text representation learning. Therefore, the conventional RNN and CNN are considered to learn the representation of text in this work.

Based on the types of input features in the deep learning model, there are two main categories of existing modeling methods: Time series modeling and content feature modeling. The time series model predicts news popularity based on temporal evolution processes of aggregated view volumes over periods. Gou et al. [10] adopted LSTM to learn sequence patterns directly from information cascading to predict the occurrence of information cascading on Twitter and Sina Weibo. Xu et al. [11] proposed a bidirectional GRU (Bi-GRU) model of the fusion attention mechanism, which can better mine the information in the resource access history and its correlation. However, Time series models have the drawback of relying on a series of historical user events. Therefore, getting the overall trend at the beginning of an online article published is difficult.

On the other hand, some recent works have demonstrated the effectiveness of content features in popularity prediction. Dou et al. [9] linked online entities with existing knowledge-based entities to propose a predictive model based on the LSTM model. Xiong et al. [28] proposed a deep learning model with news attractiveness and timeliness based on the extended model of the LDA topic model. Ghosh et al. [29] proposed a novel transfer learning approach involving sentence salience prediction as an auxiliary task. Coupled with a BERT-based neural model exceeds normalized discounted cumulative gain (NDCG) values of 0.8 for proactive sentence-specific popularity forecasting. Omidvar et al. [30] constructed the similarity, semantic, and theme modules of news titles and content to jointly determine news popularity. Compared with time-series-based methods, content features do not change over time. Therefore, content feature modeling is more reliable in the early stages of popularity prediction. However, such methods do not take advantage of the temporal evolution of popularity.

In order to take advantage of both time series and content features, multi-feature fusion was applied in some works. Liao et al. [12] proposed a deep learning model that integrates time series, text features, and meta-features and leveraged the attention mechanism to combine the three parts as output. Saeed et al. [13] proposed a hybrid deep learning model to predict the early popularity of network security news. LSTM modeled the time propagation pattern, CNN learned semantic features and deep neural network (DNN) learned other auxiliary features in the model. Fan et al. [14] proposed a neural model to predict news popularity by learning news embedding from global, local, long-term and short-term factors. Although the multi-feature fusion method uses the advantages of various features, how to represent the multiple features effectively and flexibly needs to be further studied.

In addition, the issue of predicting the popularity of news on social networks has received widespread attention. These studies are mainly based on graph neural networks, which effectively model information cascade patterns to capture predictive factors for more accurate information popularity prediction [31–33]. However, these works mainly considered the popularity prediction task via path or discrete graph modeling and are more applicable to social networks.

3 The Proposed Model

In this study, a novel MFDF model has been developed for news popularity prediction. The model consists of three main modules: time series modeling, content feature modeling, and attention fusion. In the time series modeling part, taking historical user event feedback sequence $\{v_1, v_2, \ldots, v_t\}$ as input, LSTM is adopted to get the long-term growth trends of news on the time series, and attention-based CNN is applied to capture the short-term fluctuations of news. In the content feature modeling part, one-dimensional (1-D) CNN is applied to extract the semantics of news headlines, the semantic

representation of news content is obtained by utilizing the LDA and HAN, and the high-order interaction relationship between meta-features is modeled by the FM method. In the attention fusion module, the outputs of each module are dynamically integrated through the temporal attention fusion layer. Finally, the probability distribution of news popularity is output through the fully connected (FC) layer. The framework of MFDF is shown in Fig. 1.

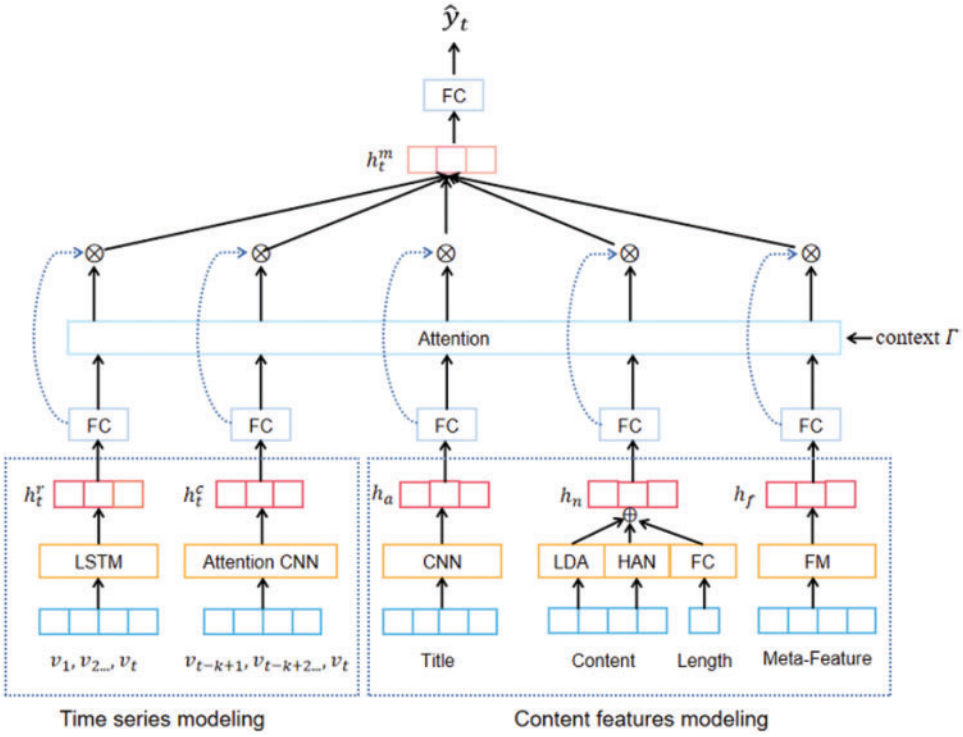

Figure 1: The framework of MFDF

3.1 Time Series Modeling

In the dataset constructed in this paper, the user's feedback events include the number of "likes," "views," and "comments." The model takes all user behaviors on the time series as a feedback vector v_t.

3.1.1 The Temporal Evolution Modeling Based on LSTM

LSTM is applied to time series modeling to capture the long-term trends of news popularity on time series. The hidden state in the LSTM model contains all historical information, so there is no need to make specific assumptions about the form of historical trends. More importantly, the memory unit can save, read, reset, and update long-distance historical information, ensuring the model can capture long-term dependencies. Therefore, the LSTM model is exploited to capture news popularity's long-term growth trends. The calculation process is shown in Eq. (1).

$$\begin{cases} f_t = sigmod\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \\ i_t = sigmod\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \\ o_t = sigmod\left(W_o \cdot [h_{t-1}, x_t] + b_o\right) \\ \widetilde{c}_t = tanh\left(W_c \cdot [h_{t-1}, x_t] + b_c\right) \\ c_t = f_t \times c_{t-1} + i_t * \widetilde{c}_t \\ h_t^r = o_t \times tanh\left(c_t\right) \end{cases} \tag{1}$$

where i_t, f_t, o_t, c_t and $\widetilde{c}_t$ represent the information of the input gate, forget gate, output gate, memory cell, and candidate memory cell; W_f, W_i, W_o and W_c represent the recursive connection weights of their corresponding thresholds; b_i, b_f, b_o and b_c are bias parameters of the input gate, forget gate, output gate, and candidate memory cell.

Then, the feedback vector $x_t = \{v_1, v_2, \ldots, v_t\}$ of each time slot is fed into the LSTM model. Finally, the output vector h_t^r represents the long-term trends of news popularity over time.

3.1.2 The Short-Term Fluctuations Modeling Based on CNN

In addition, news popularity may fluctuate in the short term due to external factors. However, the triggers of external influences are complex and most difficult to predict. Therefore, this work does not consider the complex stimuli of external forces and only obtains short-term fluctuations in news popularity based on the time series itself. As shown in Fig. 2, the overall popularity of news is a long-term trend. In contrast, short-term popularity is influenced by external factors that cause fluctuations in the popularity curve to rise or fall.

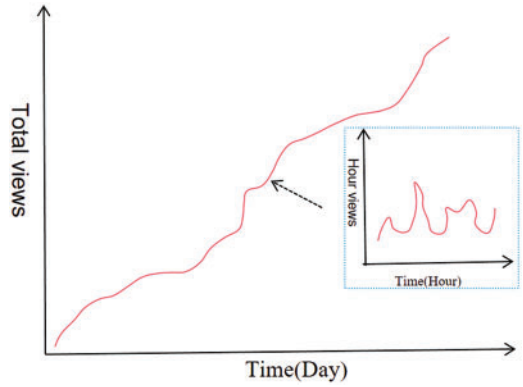

Figure 2: The short-term fluctuations of news popularity

In existing research, 1-D CNN is the best way to capture these fluctuations, so 1-D CNN is applied to capture short-term volatility in this study. Meanwhile, the fluctuations may exist in multiple time slots under the influence of external factors. Motivated by literature [12], this research employs attention-based CNN and exploits multiple convolution kernels of different sizes to fuse multiple convolution kernel outputs. The calculation process is shown in Fig. 3.

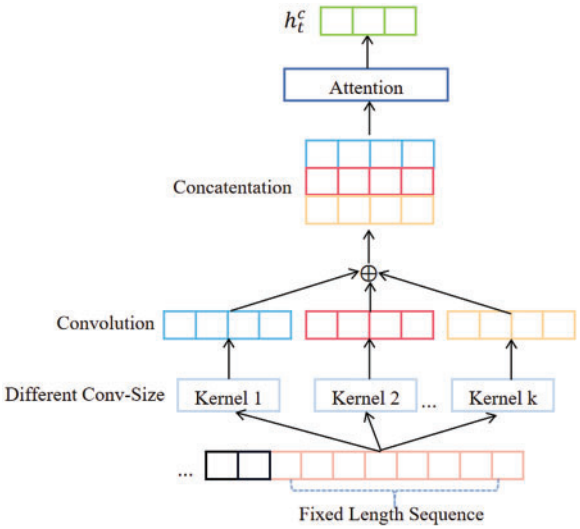

Figure 3: The process of short-term fluctuation based on CNN

Since the input of the CNN generally requires a fixed length, the time series inputs before the moment t are specified to a sub-sequence $\{v_{t-k+1}, v_{t-k+2}, \ldots, v_t\}$ of length k, then its output is also a sequence $c = \{c_{t-k+1}, c_{t-k+2}, \ldots, c_t\}$ of length k. Because time series features have different importance in the time dimension, some sub-sequences have more importance than others. Therefore, the attention mechanism is applied to merge $\{c_{t-k+1}, c_{t-k+2}, \ldots, c_t\}$ to increase the influence of important time series features and reduce the effect of non-important features. The output vector fusion method based on the attention mechanism is as follows:

$$\begin{cases} a_i^c = Q_i^c \times tanh\left(\sum_{i=1}^{k} W_j^c c_{t-k+i} + b^c\right) \\ \tilde{a}_i^c = \dfrac{exp\left(a_i^c\right)}{\sum_{j=1}^{k} exp\left(a_j^c\right)} \\ h_t^c = \sum_{i=1}^{k} \tilde{a}_i^c c_{t-k+i} \end{cases} \tag{2}$$

where $\tilde{a}_i^c$ is the attention weight of the convolution output; h_t^c is the final output of the 1-D CNN; t represents a moment in the time series; k represents the number of convolution kernels and a fixed time series length before the moment t; c_{t-k+i} represents the i-th convolution result in the output sequence c; The Q^c, b^c and W^c are learnable parameters to achieve attention scores. Q^c and W^c are weight parameters, b^c is a bias parameter.

3.2 Content Features Modeling

The content features of the news articles, such as the headline, content, and other meta-features, largely determine the popularity of the news. Therefore, it is crucial to represent this information effectively.

3.2.1 Headline Semantic Representation

In the semantic representation of news headlines, this paper adopts a typical 1-D CNN network structure [15], which can extract potential pattern features from a short text. First, a news headline is

represented as a word embedding matrix $T_{1:n} = [w_1, w_2, w_3, w_4 \ldots w_n] \in \mathbb{R}^{d \times n}$, where w_i represents the i-th word in the headline, n represents the number of words, and d represents the dimension in which the word is embedded. Then the matrix $T_{1:n}$ is convolved using convolution kernels $H \in \mathbb{R}^{d \times q}$, where $q = 3$ ($q < n$) is the window size. For each sub-matrix $T_{i:i+q-1}$, the CNN can get a feature representation o_i:

$$o_i = f\left(H * T_{i:i+q-1} + b\right) \tag{3}$$

where f is the Rectified Linear Unit (ReLu) activation function, $H * T_{i:i+q-1}$ represents the convolution operation, and b is the bias term. By all the convolution operations in the matrix, the feature sequences can be expressed as follows:

$$o = \left[o_1, o_2, o_3 \ldots, o_{n-q+1}\right]^T \tag{4}$$

Although the number of connections in the network is reduced after the convolution operation, the dimension of the features is still too high, which can easily cause overfitting, so the max-pooling layer is employed to identify the most significant features further.

$$\tilde{o} = max\left\{o_1, o_2, o_3 \ldots, o_{n-q+1}\right\} \tag{5}$$

Next, m filters with the same window size are used for convolution and max-pooling as above. Each operation can get a feature representation $\widetilde{o}$. As a result, all the features are concatenated to get the final representation of the news headlines:

$$h_t = [\tilde{o}_1, \widetilde{o}_2, \widetilde{o}_3 \ldots, \widetilde{o}_m]^T \tag{6}$$

3.2.2 News Content Semantic Representation

The online news article is usually long text that determines the popularity of the news. This paper divides the semantic representation of news content into three parts.

First, due to the great success of neural networks in the natural language process, this paper adopts the HAN [16] for modeling the news content feature. The HAN can divide text features into two levels, one is a word-level representation, and the other is a sentence-level representation. Both word-level and sentence-level encoders are Bi-GRU networks. Let A denote the news article's content. Through the processing of HAN, a news article's semantics can be expressed as h_h.

$$h_h = HAN(A) \tag{7}$$

Second, the LDA [17] topic model is trained on the whole dataset, based on which any news article's topic probabilistic representation can be obtained. Using LDA, the semantic representation of news content would be further enriched. Then, the topic representation is regarded as part of the news content semantic representation as follows:

$$h_p = LDA(A) \tag{8}$$

Third, the length of news content affects the users' activity time and has a particular impact on the popularity of the news. The news content is divided into sub-sequences with a fixed length, each containing 50 words. Subsequently, similar to the processing of word embedding, each sub-sequence is represented as a randomly initialized low-dimensional vector e_l. Finally, through a fully connected layer, the final news content length representation is expressed as follows:

$$h_l = tanh\left(W_l e_l + b_l\right) \tag{9}$$

where W_l, e_l and b_l are parameters that need to be trained.

Finally, the vectors h_h, h_p and h_l are concatenated and fed into a fully connected layer to get the final content semantic representation h_n. The fully connected layer employs ReLu as an activation function.

$$h_n = FC\left(\left[h_h; h_p; h_l\right]\right) \tag{10}$$

3.2.3 Meta Features Representation

Besides headlines and content, news article also has many meta-features, such as author, number of followers, location, and publish time, which affect the popularity of news to some extent. Some generic methods, such as One-hot, may make the input dimension huge. In addition, there is an interaction among news meta-features, such as author and number of followers may influence news popularity. Therefore, it is necessary to learn meta-feature representation to reduce the size of input dimensions and capture higher-order interactions.

This paper adopts FM [18] to deal with interactions between meta-features, which use pairwise feature interactions as the inner product of potential vectors to solve the problem of low and high-order feature combination extraction. The discrete features are embedded as dense vectors first. Although FM can model high-order feature interactions in theory, it has high complexity in practice. Therefore, this work adopts only order-2 feature interaction. The core process of the FM layer is expressed as follows:

$$\begin{cases} \left\langle v_i, v_j \right\rangle = \sum_{f=1}^{D} v_{i,f} v_{j,f} \\ h_f = w_0 + wx + \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} \left\langle v_i, v_j \right\rangle x_i x_j \end{cases} \tag{11}$$

where the first half of the Eq. (11) is a linear combination of features, and the second half is a cross-combination of features; $w_0 \in \mathbb{R}^F$ is a global bias; $x \in \mathbb{R}^L$ is the concatenated result of the meta-features after the embedding operation; L is the size of x; $w \in \mathbb{R}^{F \times L}$ is the weight of x; x_i and x_j are elements of x; $\left\langle v_i, v_j \right\rangle$ is the dot product of two vectors of size D in matrices; v_i and v_j represent the cross-weighting of the features i and j; $v_i \in \mathbb{R}^{F \times D}$ is the hidden vector of the i-th feature; $D \in \mathbb{N}_0^+$ is a hyperparameter that defines the dimension of factorization; h_f is the latent output of the FM layer, which is part of the news content semantic representation.

3.3 Attentive Fusion Unit

In the previous article, through time series modeling techniques, the output vector h_t^r represents the historical growth pattern of news and the output vector h_t^c represents the short-term fluctuation pattern. Through the content feature modeling techniques, the semantic representation of news headlines h_t, the semantic representation of news content h_n and the semantic representation of news meta features h_f are obtained. For the output of each module, the most direct way is to concatenate all of the outputs and feed the concatenation results into the fully connected network layer for prediction. However, this approach means that the weight vectoris fixed. As mentioned above, in the early stage of news published, it is not easy to get the overall growth trend of news popularity by utilizing time series modeling. Therefore, the prediction in the early stages of news published relies heavily on content feature modeling. As time passes, the information available to the model on the time series gradually increases, and the observed popularity gets closer to the overall popularity. Hence, time series modeling plays different roles in different time stages. Based on the above analysis, this paper adopts an attention mechanism to integrate multiple features flexibly. The value of the attention weight is calculated by h_t^r, h_t^c, h_a, h_n, h_f and temporal context vector Γ. The calculation process of vector Γ is shown in Fig. 4.

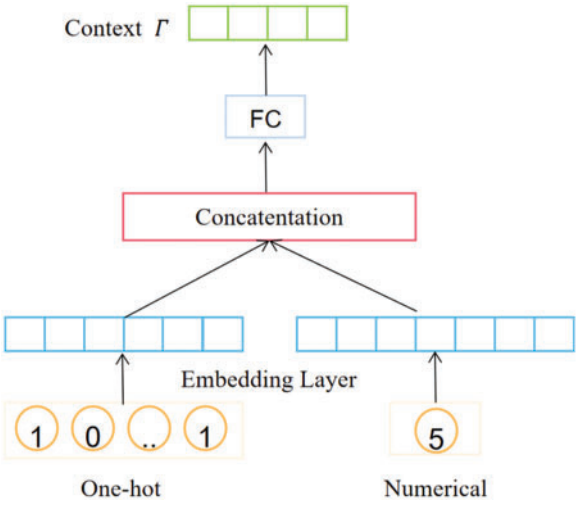

Figure 4: The calculation process of temporal context vector Γ

The time context vector Γ consists of the period properties of the time, the time interval of t and the publish time. The period properties are One-hot features, representing each hour of the day and measuring the impact of different times on popularity. The time interval is a numerical feature measured in 0.5 h. First, an embedding layer is exploited to embed these features into homologous dense vectors. Then the embedding vectors are concatenated and fed into a fully connected layer to get the time context vector Γ.

$$\begin{cases} t_c = [Embed(hour); Embed(interval)] \\ \Gamma = tanh(W_n t_c + b_n) \end{cases} \tag{12}$$

where *Embed* represents the embedding operation, W_n and b_n are the parameters that need to be trained in the model, and t_c is the result of concatenation.

Then, the h_t^r, h_t^c, h_a, h_n and h_f are fed into the fully connected layers to get the aligned vectors $\hat{h}_t^r$, $\hat{h}_t^c$, $\hat{h}_t^a$, $\hat{h}_t^n$ and $\hat{h}_t^f$. The attention weight of each module is calculated as follows:

$$\begin{cases} \alpha_i^m = q_i^m tanh\left(\sum_{j\in\{r,c,a,n,f\}} W_j^m \hat{h}_t^j + W_t^m \Gamma + b^m\right) \\ \tilde{\alpha}_i^m = \dfrac{exp\left(\alpha_i^m\right)}{\sum_{k\in\{r,c,a,n,f\}} exp\left(\alpha_k^m\right)} \end{cases} \tag{13}$$

where q^m, W^m, b^m are learnable parameters to achieve attention scores, and $\tilde{\alpha}_i^m$ is the attention weight assigned to the outputs of each module.

3.4 The Output Layer

In the output layer, the attention weight is applied to combine the outputs of each module into h_t^m, and then the h_t^m is fed into a fully connected layer and softmax layer to get a probability distribution

$P_t = \{p_t(c_1), p_t(c_2), \ldots, p_t(c_m)\}$. The max probability is the prediction result $\hat{y}_t$. The calculation process of the output stage is as follows:

$$\begin{cases} h_t^m = \sum_{i\in\{r,c,a,n,f\}} \tilde{a}_i^m \hat{h}_t^i \\ P_t = softmax\left(W_h \times h_t^m + b_h\right) \\ \hat{y}_t = argmax\left(P_t\right) \end{cases} \tag{14}$$

3.5 Model Training

Supposing the true popularity of news article a is c_a at time slot t, the single-step cross-entropy loss can be defined as follows:

$$L_t = -\sum_{i=1}^{m} y_i \log(p_t(c_a)) \tag{15}$$

Then the overall loss of the entire time series of news article a can be defined as:

$$\mathcal{L}_{total} = \sum_t \mathcal{L}_t \tag{16}$$

Because the news popularity prediction task is probably susceptible to severe class imbalance. Therefore, this work adopts re-weighting [34] technology to adjust each class weight. Specifically, a class's effective size is defined as:

$$E_n = \frac{1-\beta^n}{1-\beta} \tag{17}$$

where n is the actual number of samples in a class, and β is a hyperparameter whose value is usually close to 1.

Accordingly, the loss function $\mathcal{L}_{total}$ based on class-balanced can be expressed as:

$$\mathcal{L}_{total} = \frac{1}{E_{n_{y_i}}} \sum_t \mathcal{L}_t \tag{18}$$

where n_{y_i} is the actual number of samples in class y_i.

In practice, predicting the popularity of news in the early stage is more valuable. This research exploits a time decay factor for the single-step loss to put more effort into optimizing prediction performance at an early stage:

$$\mathcal{L}_{total} = \frac{1}{E_{n_{y_i}}} \sum_t D(\Delta t)\mathcal{L}_t \tag{19}$$

where $D(\Delta t)$ is a monotonous and nonincreasing function of the time interval Δt between t and the publishing time.

Since Newton's law of cooling measures the exponential form of decay between temperature and time, Newton's law of cooling is often used in the decay process of item heat in some recommendation algorithms [35]. This study employs a time decay factor based on Newton's law of cooling to ensure the decay rate of $D(\Delta t)$ will get smaller and smaller with time goes by. The decay factor $D(\Delta t)$ is expressed as:

$$D(\Delta t) = e^{-\alpha \Delta t} \tag{20}$$

where α is a hyperparameter for controlling the decay rate; Δt is the number of time slots from the release time to t.

4 Experiment

This section examines the effectiveness of MFDF by comparing it with several competitive baselines and conducting extensive experiments for evaluating model prediction performances in different stages.

4.1 Dataset

The data come from a widely used mobile news platform Toutiao (www.toutiao.com). Compared with other news platforms, Toutiao's users are more active, with more user interactive behaviors, which can provide richer data for this study. The news articles come from the five items at the top of the homepage, and the publish date is between November 5, 2020, and June 15, 2022. The collected data comprises multiple-dimensional information such as news headlines, author, publish time, news content, likes, and comments. For example, we randomly selected a piece of news with the headline "China's GDP exceeded 100 trillion yuan for the first time." This news article was posted by CCTV News at 15:08 on January 18, 2021, with more than 310,000 views, 4295 comments and 11000 likes, which spread on the homepage of Toutiao for 4 h and 20 min. In addition, CCTV News has a hundred million followers. The data is collected every 15 min to fully obtain the sequence features over time. This paper divides the overall popularity of news articles into three classes, hot (more than 200000 views), cold (less than 50000 views), and normal (otherwise). This paper describes the amount of "view," "comment," and "like" actions per 15 min of each article as macro time series. To get sufficient time series information, some news articles' time series with fewer than eight are removed. In addition, this paper clips these time series before the observed popularity reaches 80% of overall popularity. Finally, The dataset comprises 12108 articles. The division of the training, test, and validation set is shown in Fig. 5.

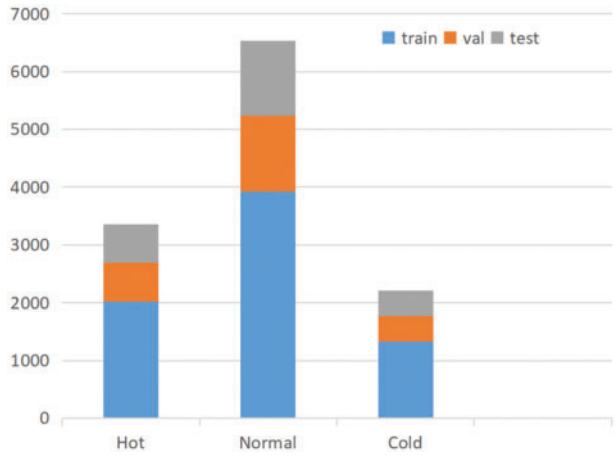

Figure 5: Data partitioning

4.2 Experiment Setup

In the process of time series modeling, the cell size of LSTM was empirically set as 512. In the attention-based CNN layer, three kernels of sizes 3, 5, and 7 are used to perform convolution. The number of each kind of kernel is set as 128. The window size k of CNN is 1 h.

For content features learning, because 98.2% of news headlines in the dataset were less than 20 words after removing stop words, this paper sets the maximum length of the headlines as 20 in the experiment and fills in shorter sequences or truncates longer sequences. In addition, the maximum length of news content is set as 500. In the news headline representation stage, the embedding layer adopts a 100-dimensional word vector pre-trained by Word2vec [36]. The window size q of CNN is 3, and the number of kernels is set as 60. For news content representation learning, this work starts with pre-trained HAN and LDA for news content representation and fine-tuning parameters with the dataset. For meta-feature representation learning, the size of embedding dimensionality was empirically set as 50, and the dimensionality of factorization D was set as 3 based on the experience of literature [37]. All of the fully connected layers have a suitable hidden size of 512. Through multiple experiments, the values of α and β are set as 0.1 and 0.999 in the decayed loss function. To reduce dimensionality and avoid overfitting, this paper adopts the Xavier initialization and Adam optimizer for parameters learning, and a dropout of 0.2 is applied after the fully connected layers and RNN layers for regularization.

At last, a softmax layer is adopted, classifying news into different popularity levels. The model is trained for 20 epochs with a batch size of 32. Training is based on Pytorch 1.11.0 framework, and one NVIDIA A10 is used for calculation.

4.3 Evaluation Metrics

The performances of the proposed model are evaluated using accuracy and macro averaged F-Score that have been commonly used for assessing classification models. Accuracy is the ratio of correctly predicted samples to the total samples. Macro averaged F-Score is the mean of F-scores of each class which can evaluate the model's overall performance in a global sense. The formulas of macro averaged F-Score are given below:

$$\mathrm{F-Score} = \frac{1}{C}\sum_{c=1}^{C}\frac{2P_cR_c}{P_c + R_c} \tag{21}$$

where P_c denotes the percentage of all news identified as positive samples that are indeed positive ones in class c; R_c is the percentage of positive predictions that are correctly recognized in class c.

4.4 Comparison of the Proposed Model with the Baseline Models

In order to evaluate the effectiveness of the proposed model, this paper compared its performance with some state-of-the-art competitors. Their ideas are commonly used in text classification and popularity prediction. The details of these models are as follows:

Multivariate Analysis (**MA**) [38]: This model utilizes news features such as followers count, content length, and publish time and adopts the linear regression model to predict news popularity.

Hierarchical Neural Network (**HNN**) [8]: This model combines the advantages of CNN and LSTM to predict popularity. The CNN generates a distributed representation of the sentence, and the LSTM process the sequence of sentences to model the semantic relationship between sentences.

Ensemble Learning (**EL**) [6]: This model adopts LDA topic modeling and Doc2vec [39] embedding technique for getting news features and predicting news popularity through ensemble learning.

DeepFM [40]: This model combines FM with the DNN, the state-of-the-art method of meta-feature learning.

Determining the Quality of News Headlines (**DQNH**) [30]: This model determines the results of news classification by constructing the similarity, semantic, and topic modules of headlines and news content. The three-module outputs jointly determine the effects of prediction. In this model, BERT [26] transforms headlines and news content into fixed-length vectors to find the latent features of headlines and news content.

Deep Fusion of Temporal Process and Content Features (**DFTC**) [12]: This model combines time series and content features to predict news popularity. The attention layer is used to concatenate multiple representations.

Deep Temporal Propagation Patterns (**DTPP**) [13]: This model exploits deep neural networks to learn different features. The direct concatenation of CNN, LSTM, and DNN outputs determines the popularity prediction results.

Context-Aware Convolutional Neural Network (**CACNN**) [41]: This model adopts CNN for click-rate prediction, attention CNN to model time processes, and multi-layer neural networks to represent metadata.

It is evident from Table 1 that the MFDF model yields better results than the baseline models in terms of Accuracy and F-Score. (1) The MA has the worst performance. Since the MA is a feature-based linear prediction model, its performance largely depends on the extracted features, and its feature extraction is difficult. Since DeepFM contains low-order and high-order interactions of meta-features, its performance is improved. However, neither model takes advantage of the features of news text and time series, so the MA and DeepFM models yield the worst classification results in the experiment. (2) The HNN, EL, and DQNH adopt various methods to obtain the semantic representation of news texts, but there are also significant differences. The DQNH gets the semantic relationship, semantic features, and theme features of news headlines and content, which capture more comprehensive and rich news text representation, yielding the best performance. The HNN model also considers the influence of news content and headlines on popularity prediction, but HNN combines news headlines and content into one document to get a common semantic representation of both. As a result, the HNN model cannot distinguish between news headlines and content. The EL model adopts LDA for representing the news topic. Meanwhile, Doc2Vec is used to represent the news headline and content. Its essence is to extract multiple news features and then input the vectorized features into random forest (RF), logistic regression (LR) and SVM. The classification results through the voting strategy. The problem with the EL model is that although mature text representation methods are adopted, feature extraction and model training are two independent stages. Therefore, it is not easy to achieve optimal performance. None of the above models considers the temporal relationship and news meta-features. (3) The CACNN, DFTC and DTPP consider time series information. Regarding time series modeling, CACNN adopts attention CNN to process time series to capture nonlinear local information; The DFTC adds LSTM module to get the long-term growth trend of news popularity. The DTPP captures the temporal propagation pattern mode of news through LSTM. Regarding content feature modeling, the DFTC and DTPP leverage text features and meta-features of news, while CACNN only utilizes meta-features. Regarding multi-feature fusion, DFTC employs an attention mechanism to dynamically assign weights, while CACNN and DTPP directly connect feature vectors. The experimental results show that in the three models, the performance of the DFTC model is better

than that of DTPP, and the performance of the CACNN model is the worst. Regarding multi-feature fusion, the DFTC exploited an attention mechanism for dynamically assigning weights, while CACNN and DTPP directly concatenate vectors. The experimental results show that in the three models, the performance of the DFTC model is better than the DTPP, and the performance of the CACNN model is the worst. (4) Compared with them, the proposed MFDF captures the time series process and adopts more advanced text features and meta-feature representation methods. Based on these modeling techniques, the MFDF model significantly outperforms state-of-the-art methods.

Table 1: Comparison of the proposed model with the baseline models

Model	Accuracy (%)	F-score (%)	Model size (M)	Inference time (ms)
MA	68.23	65.12		
HNN	73.64	70.25	5.3	2.07
EL	71.23	70.34		
DeepFM	70.67	67.15	6.1	1.53
DQNH	75.67	74.42	9.1	1.89
DFTC	83.28	81.56	8.8	2.57
DTPP	80.13	78.46	7.5	2.18
CACNN	78.45	75.65	7.3	2.12
MFDF (ours)	**85.16**	**83.71**	**10.2**	**3.03**

Meanwhile, this work analysis the number of parameters and inference time in the experiments. Since the MA and EL models use classical learning methods, they are not compared with other deep learning models. As for the deep learning network, the model size is the storage of parameters. Under the experiment setting of the proposed model, the total number of parameters is 10.2M. Since the proposed model incorporates multiple features, it has more parameters than the comparison models. For comparing the inference time cost of the MFDF model with the comparison models, this study conducts prediction experiments with batch size 32. In the experiment, the MFDF model takes 3.03 ms for one-step prediction and does not significantly increase the inference time.

4.5 Ablation Study

This paper carries out the corresponding ablation experiments to verify the influence of different sub-modules on the overall task performance of the MFDF. As shown in Table 2, the time series sub-module (TS), content features sub-module (FS), and temporal attention mechanism (AM) impact the overall model performance.

Table 2: Ablation model performance

TS	FS	AM	Accuracy (%)	F-score (%)
√			74.48	72.39
	√		77.53	76.29
	√		82.83	81.32
√	√	√	**85.16**	**83.71**

In single-module experiments, TS and FS only adopt time series or content features to make predictions, and the model performance is relatively poor. The performance of using only the FS module is higher than DQNH, which shows the effectiveness of the content feature modeling method in the FS module. TS+FS combines time series with content features modeling through vector concatenation, significantly improving performance. It shows that the time series and the content features are complementary but lack the flexibility to handle the dynamic growth of the time series. Ultimately, the combination of all modules can further improve the prediction performance. Time series and content features can be assigned dynamic weights through the time attention module at different stages. The performance differences between the comparison model and the module combination are shown in Fig. 6.

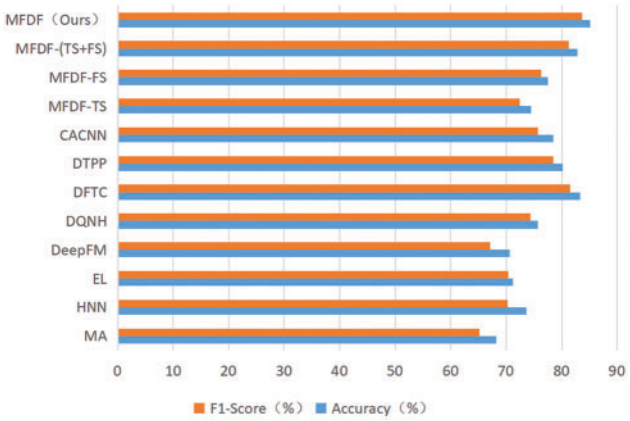

Figure 6: The performance comparison

4.6 The Performance in Early Stage

Predicting overall popularity in the early stage of news articles published is more valuable in practical applications. Therefore, this paper evaluates four models considering time series features: DTPP, CACNN, DFTC, and MFDF. Fig. 7 shows the average performance of the first 5 h after news articles are published. As shown in Fig. 7, the proposed model's prediction performance improved significantly in the early stage. The CACNN has poor early prediction performance because it adopts only meta-features for prediction in content feature modeling. The DTPP adopts news text and meta-features for content feature modeling, which can capture richer information in the early stage, so the prediction performance is improved. The DFTC adopts the temporal attention mechanism to dynamically fuse time series, content features and meta-features, which can get desirable prediction performance when the time series lacks sufficient information. Attention fusion can ensure that content feature modeling plays a more significant role in the early stage. The MFDF model adds more content features and considers the higher-order interaction of meta-features. Meanwhile, the loss function based on Newton's cooling law helps the model to invest more effort in optimizing the prediction performance in the early stage. In addition, because the observed news popularity gets closer and closer to the overall popularity over time, the model can achieve ideal performance even if the weight of content features is reduced in the later stage.

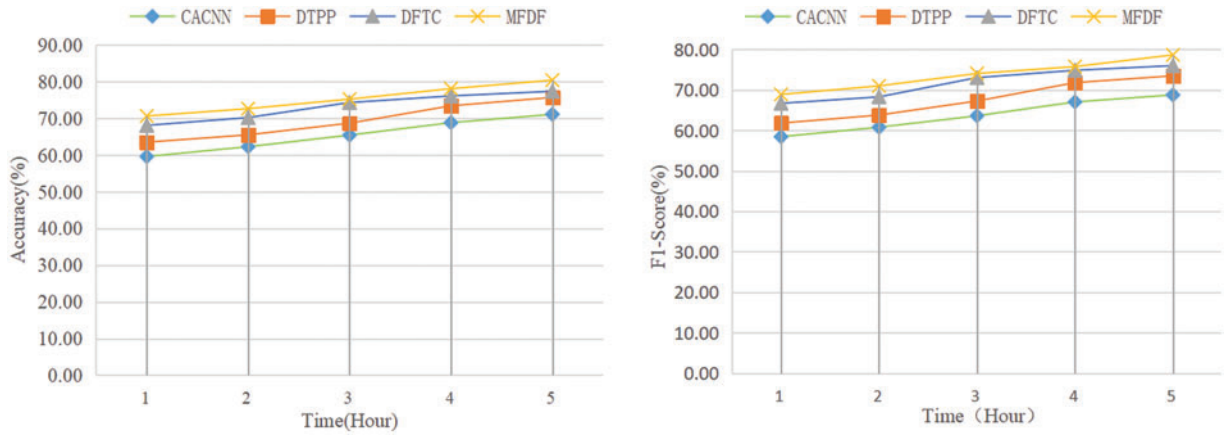

Figure 7: The model performance in the early stage

4.7 Attention Fusion Effect

This paper randomly selects a news article as a case study to study the essential role of the fusion layer in the experiment. First, the time series of the news are divided into five stages on average. Then, we average the attention weight for each stage and aggregate its views. As shown in Fig. 8, the heat map represents attentive weights $\widetilde{a}_r^m$, $\widetilde{a}_c^m$, $\widetilde{a}_a^m$, $\widetilde{a}_n^m$ and $\widetilde{a}_f^m$. The darker the color is, the bigger the weight is. It can be seen that each module has a matching contribution at the beginning. The attention weight assigned by the model to $\hat{h}_t^r$ gradually increases. Thus LSTM model plays a vital role in prediction at the later period. In addition, the weight assigned by the model to the content features is gradually reduced. Therefore, the content features play a more significant role in the early stage. The line chart shows the short-term fluctuations of this news article in Fig. 8. It can be seen that during some phases of the time series, there are significant fluctuations in view amount. The output of the attention CNN model is given a higher weight during the fluctuation phase, which means it effectively captures such local fluctuations.

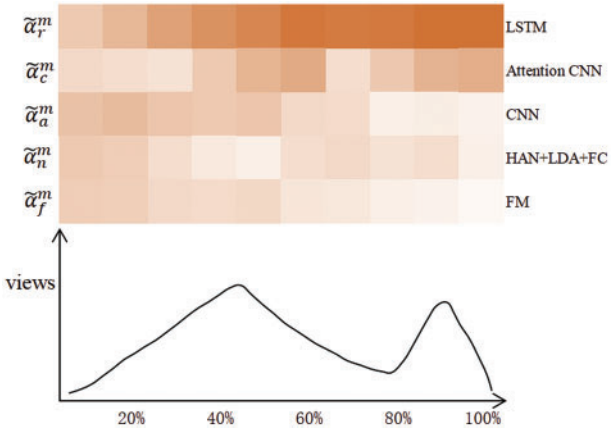

Figure 8: The attentive heat map

4.8 The Effectiveness of FM Method

In order to verify the effectiveness of the FM method in enhancing meta-feature representation, this paper also adopts embedding techniques for embedding meta-feature into homologous dense vectors and exploits a fully connected layer to get the final representation. As shown in Table 3, the performance of predictions can be further improved using the FM method.

Table 3: Comparison of meta-feature representation method

Method	Accuracy (%)	F-score (%)
FM	**85.16**	**83.71**
Concatenation	84.12	81.23

4.9 Comparison of Loss Functions

To verify the effectiveness of the loss function, the experiments employ a loss function without a time decay factor and a log time decay factor-based loss function. The loss function without the time decay factor is shown in Eq. (18). The logtime decay factor is a monotonic non-incrementing function designed by the DFTC model [12], and its formula is as follows:

$$D(\Delta t) = \left\lceil \log_{\gamma}(\Delta t + 1) \right\rceil^{-1} \tag{22}$$

where $\lceil . \rceil$represents up rounding operator; $\gamma > 1$ is a hyper-parameter for controlling the decay rate. The value of γ is set as 12 based on the design of the DFTC.

As shown in Table 4, the loss function based on the time decay factor helps to improve the model prediction performance. Among them, the time decay factor based on Newton's cooling law performed better in the experiment.

Table 4: Comparison of loss functions

Loss function	Accuracy (%)	F-score (%)
Newton's law of cooling decay factor	**85.16**	**83.71**
Without decay factor	83.76	82.91
Log decay factor	84.65	83.13

5 Conclusion

This paper proposes a novel neural network model named Multi-Feature Dynamic Fusion (MFDF) for news popularity prediction, which fully uses time series and content features, and improves prediction performance through effective representation and dynamic fusion of various features. The extensive experiments on real-world news datasets have confirmed that the proposed model significantly outperforms state-of-the-art methods, demonstrating the validity and superiority of the proposed model. This work will further improve in the following aspects in the future: (1) The data used in the experiment are collected from Toutiao. It would be beneficial to validate the proposed method with different news corpora. (2) Exploring better methods to expand news content features,

such as attractiveness and timeliness of news. (3) Explore the impact of the transformer models for semantic representation on popularity prediction performance.

Acknowledgement: We thank the anonymous reviewers for their valuable and helpful comments, which helped us improve this paper's content and presentation.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

[1] Y. Keneshloo, S. Wang, E. Han and N. Ramakrishnan, "Predicting the popularity of news articles," in *2016 SIAM Int. Conf. on Data Mining*, Siam, Tha, pp. 441–449, 2016.

[2] S. Choudhary, A. S. Sandhu and T. Pradhan, "Genetic algorithm based correlation enhanced prediction of online news popularity," in *2017 Int. Conf. on Computational Intelligence in Data Mining*, Singapore, Springer, pp. 133–144, 2017.

[3] E. A. Khan, G. Worah, M. Kothari, Y. H. Jadhav and A. V. Nimkar, "News popularity prediction with ensemble methods of classification," in *IEEE 2018 9th Int. Conf. on Computing, Communication and Networking Technologies (ICCCNT)*, Bengaluru, India, pp. 1–6, 2018.

[4] S. Abbar, C. Castillo and A. Sanfilippo, "To post or not to post: Using online trends to predict popularity of offline content," in *Proc. Hypertext and Social Media*, New York, NY, USA, pp. 215–219, 2018.

[5] M. J. Tsai and Y. Q. Wu, "Predicting online news popularity based on machine learning," *Computers and Electrical Engineering*, vol. 102, no. 8, 108198, 2022.

[6] F. Long, M. Xu, Y. Li, Z. Wu and Q. Ling, "XiaoA: A robot editor for popularity prediction of online news based on ensemble learning," in *Intelligence Science II: Third IFIP TC 12 Int. Conf.*, Beijing, China, pp. 340–350, 2018.

[7] Y. Yang, Y. Liu, X. Lu, J. Xu and F. Wang, "A named entity topic model for news popularity prediction," *Knowledge-Based Systems*, vol. 208, no. 3, 106430, 2020.

[8] X. Guan, Q. Peng, Y. Li and Z. Zhu, "Hierarchical neural network for online news popularity prediction," in *2017 Chinese Automation Congress (CAC)*, Jinan, China, pp. 3005–3009, 2017.

[9] H. Dou, W. X. Zhao, Y. Zhao, D. Dong, J. Wen *et al.,* "Predicting the popularity of online content with knowledge-enhanced neural networks," in *24th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, London, UK, 2018.

[10] C. Gou, H. Shen, P. Du, D. Wu, Y. Liu *et al.,* "Learning sequential features for cascade outbreak prediction," *Knowledge and Information Systems*, vol. 57, no. 3, pp. 721–739, 2018.

[11] Y. Xu and G. J. Liu, "Bi-GRU content popularity prediction algorithm based on attention mechanism," *Electronic Measurement Technology*, vol. 45, no. 3, pp. 54–60, 2022.

[12] D. Liao, J. Xu, G. Li, W. Huang, W. Liu *et al.,* "Popularity prediction on online articles with deep fusion of temporal process and content features," in *Thirty-Third AAAI Conf. on Artificial Intelligence*, Hawaii, USA, pp. 200–207, 2019.

[13] R. Saeed, H. Abbas, S. Asif, S. Rubab, M. M. Khan *et al.,* "A framework to predict early news popularity using deep temporal propagation patterns," *Expert Systems with Applications*, vol. 195, no. 6, 116496, 2022. https://doi.org/10.1016/j.eswa.2021.116496

[14] S. Fan, C. Lin, H. Li and Q. Zou, "News popularity prediction with local-global long-short-term embedding," in *22nd Int. Conf. on Web Information Systems Engineering (WISE 2021)*, Melbourne, Australia, pp. 79–93, 2022.

[15] Y. Chen, "Convolutional neural network for sentence classification," M.S. dissertation, University of Waterloo, Waterloo, Canada, 2015.

[16] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola *et al.,* "Hierarchical attention networks for document classification," in *2016 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, USA, pp. 1480–1489, 2016.
[17] D. M. Blei, A. Y. Ng and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, no. 2, pp. 993–1022, 2003.
[18] S. Rendle, "Factorization machines," in *2010 IEEE Int. Conf. on Data Mining*, Sydney, Australia, pp. 995–1000, 2010.
[19] S. Rajagopal, A. Kadan, M. G. Prappanadan and L. V. Lakshmanan, "Online news popularity prediction before publication: Effect of readability, emotion, psycholinguistics features," *IAES International Journal of Artificial Intelligence*, vol. 11, no. 2, pp. 539–545, 2022.
[20] W. Stokowiec, T. Trzcinski, K. Wołk, K. Marasek and P. Rokita, "Shallow reading with deep learning: Predicting popularity of online content using only its title," in *Foundations of Intelligent Systems: 23rd Int. Symp.*, Warsaw, Poland, pp. 136–145, 2017.
[21] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179–211, 1990. https://doi.org/10.1207/s15516709cog1402_1
[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[23] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997. https://doi.org/10.1109/78.650093
[24] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk *et al.,* "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, pp. 1724–1734, 2014.
[25] Y. N. Dauphin, A. Fan, M. Auli and D. Grangier, "Language modeling with gated convolutional networks," in *Int. Conf. on Machine Learning*, New York, USA, PMLR, pp. 933–941, 2017.
[26] J. D. M. C. Kenton and L. K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *17-rd Annual Conf. of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, Minneapolis, USA, pp. 4171–4186, 2019.
[27] A. Rastogi, Q. Liu and E. Cambria, "Stress detection from social media articles: New dataset benchmark and analytical study," in *2022 Int. Joint Conf. on Neural Networks (IJCNN)*, Padua, Italy, pp. 1–8, 2022.
[28] J. Xiong, L. Yu, D. Zhang and Y. Leng, "DNCP: An attention-based deep learning approach enhanced with attractiveness and timeliness of News for online news click prediction," *Information & Management*, vol. 58, no. 2, pp. 103428, 2021.
[29] S. Ghosh Roy, A. Padhi, R. Jain, M. Gupta and V. Varma, "Towards proactively forecasting sentence-specific information popularity within online news documents," in *Proc. Hypertext and Social Media*, New York, NY, USA, pp. 11–20, 2022.
[30] A. Omidvar, H. Pourmodheji, A. An and G. Edall, "A novel approach to determining the quality of news headlines," in *Proc. NLPinAI*, Berlin, German, Springer, pp. 227–245, 2021.
[31] X. Q. Jia, J. X. Shang, D. J. Liu, H. D. Zhang and W. C. Ni, "HeDAN: Heterogeneous diffusion attention network for popularity prediction of online content," *Knowledge-Based Systems*, vol. 254, no. 6380, 109659, 2022. https://doi.org/10.1016/j.knosys.2022.109659
[32] C. Zhong, F. Xiong, S. R. Pan, L. Wang and X. Xiong, "Hierarchical attention neural network for information cascade prediction," *Information Sciences*, vol. 622, no. 65–68, pp. 1109–1127, 2023. https://doi.org/10.1016/j.ins.2022.11.163
[33] C. Yang, P. Bao, R. Yan, J. Li and X. Li, "A graph temporal information learning framework for popularity prediction," in *Companion Proc. of the Web Conf. 2022*, Lyon, France, pp. 239–242, 2022.
[34] Y. Cui, M. Jia, T. Lin, Y. Song and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proc. CVPR*, Long Beach, CA, USA, pp. 9268–9277, 2019.
[35] X. Han, Z. Wang and H. J. Xu, "Time-weighted collaborative filtering algorithm based on improved mini batch K-means clustering," *Advances in Science and Technology*, vol. 105, pp. 309–317, 2021.

[36] S. Li, Z. Zhao, R. Hu, W. Li, T. Liu *et al.,* "Analogical reasoning on Chinese morphological and semantic relations," in *56th Annual Meeting of the Association for Computational Linguistics (ACL)*, Melbourne, Australia, pp. 138–143, 2018.

[37] S. Rendle, "Factorization machines with libFM," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 3, no. 3, pp. 1–22, 2012.

[38] C. Liu, W. Wang, Y. Zhang, Y. Dong, F. He *et al.,* "Predicting the popularity of online news based on multivariate analysis," in *2017 IEEE Int. Conf. on Computer and Information Technology (CIT)*, Helsinki, Finland, pp. 9–15, 2017.

[39] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Int. Conf. on Machine Learning*, New York, USA, PMLR, pp. 1188–1196, 2014.

[40] S. H. Guo, R. Tang, Y. Ye, Z. Li and X. He, "DeepFM: A factorization-machine based neural network for CTR prediction," in *26th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Melbourne, Australia, pp. 1725–1731, 2017.

[41] H. Gao, D. Kong, M. Lu, X. Bai and J. Yang, "Attention convolutional neural network for advertiser-level click-through rate forecasting," in *Proc. of the 2018 World Wide Web Conf.*, Lyon, France, pp. 1855–1864, 2018.