AI Open 5 (2024) 1–9

Contents lists available at ScienceDirect

AI Open

journal homepage: www.keaipublishing.com/en/journals/ai-open

Full length article

How to generate popular post headlines on social media?

Zhouxiang Fang [a,*], Min Yu [b], Zhendong Fu [a], Boning Zhang [a], Xuanwen Huang [a], Xiaoqi Tang [b], Yang Yang [a]

[a] Department of Computer Science and Technology, Zhejiang University, Hangzhou, 310013, China
[b] Huayun Information and Technology Co. Ltd., Hangzhou, 310013, China

ARTICLE INFO

Keywords:
Social media
Data mining
Headline generation

ABSTRACT

Posts, as important containers of user-generated-content on social media, are of tremendous social influence and commercial value. As an integral component of post, headline has decisive influence on post's popularity. However, the current mainstream method for headline generation is still manually writing, which is unstable and requires extensive human effort. This drives us to explore a novel research question: Can we automate the generation of popular headlines on social media? We collect more than 1 million posts of 42,447 thousand celebrities from public data of Xiaohongshu, which is a well-known social media platform in China. We then conduct careful observations on the headlines of these posts. Observation results demonstrate that trends and personal styles are widespread in headlines on social medias and have significant contribution to posts' popularity. Motivated by these insights, we present MEBART, which combines **M**ultiple preference-**E**xtractors with **B**idirectional and **A**uto-**R**egressive **T**ransformers (BART), capturing trends and personal styles to generate popular headlines on social medias. We perform extensive experiments on real-world datasets and achieve SOTA performance compared with advanced baselines. In addition, ablation and case studies demonstrate that *MEBART* advances in capturing trends and personal styles.

1. Introduction

As the Internet develops over time, social medias have become an integral part of our lives (McMillan and Morrison, 2006). Billions of user-generated-content pieces, such as opinions and experiences, are presented on social medias everyday (Saravanakumar and SuganthaLakshmi, 2012). And posts are the basic containers of them (Morrison et al., 2013). Fig. 1 shows a post on Xiaohongshu,[1] a well-known social media platform in China. Due to its huge reads, some popular posts help exert a tremendous influence for a wide range of applications, including government policy propagandizing (Bradshaw et al., 2020), product marketing (De Vries et al., 2012), etc. Therefore, creating popular posts is of huge commercial and research value.

Creating a popular headline is fundamental to create a popular post. According to the literature (Lakkaraju et al., 2013; Zagovora et al., 2018), a headline, as a crucial part of a post (Meij et al., 2012), has a decisive influence on the post's popularity. A successful headline can raise the post's popularity (reads) by drawing more readers' attention. Besides, the headline summarizes the central point of a post, the quality of which might impact the readers' initial impressions.

In the past, individual users usually create headlines by themselves. However, it is hard to guarantee the popularity of the generated headlines due to the large variation in individual knowledge and personal properties. Recently, some companies (e.g., the Multi-Channel Network[2]) notice the massive market of this business opportunity and provide professional writing and ghostwriting service for clients (Gardner and Lehnert, 2016). However, user interests and social trends are always changing. Therefore,it takes extensive human effort to keep up with these changing factors. What is more, MCN's service is rather expensive for the users, hindering themselves from becoming prevalent among users. Therefore, this paper explores a novel research question: **Can we automate the generation of popular headlines on social media?**

Currently, with the advancement of the sequence-to-sequence technique in Natural Language Processing (NLP) (Vaswani et al., 2017; Devlin et al., 2018; Lewis et al., 2019), headline generation methods have been studied extensively (Liu et al., 2022; Lewis et al., 2019) and achieved a great success in a variety of scenarios, such as news writing (Ao et al., 2021), online ads (Kanungo et al., 2022), and paper

* Corresponding author.
E-mail address: zhouxiangfang@zju.edu.cn (Z. Fang).

[1] http://xhslink.com/rBi59x

[2] The Multi-Channel Network (MCN) is any entity or organization which either partners with content creators or directly produces a variety of distinctive content.

https://doi.org/10.1016/j.aiopen.2023.12.002
Received 30 November 2023; Received in revised form 8 December 2023; Accepted 11 December 2023
Available online 16 December 2023

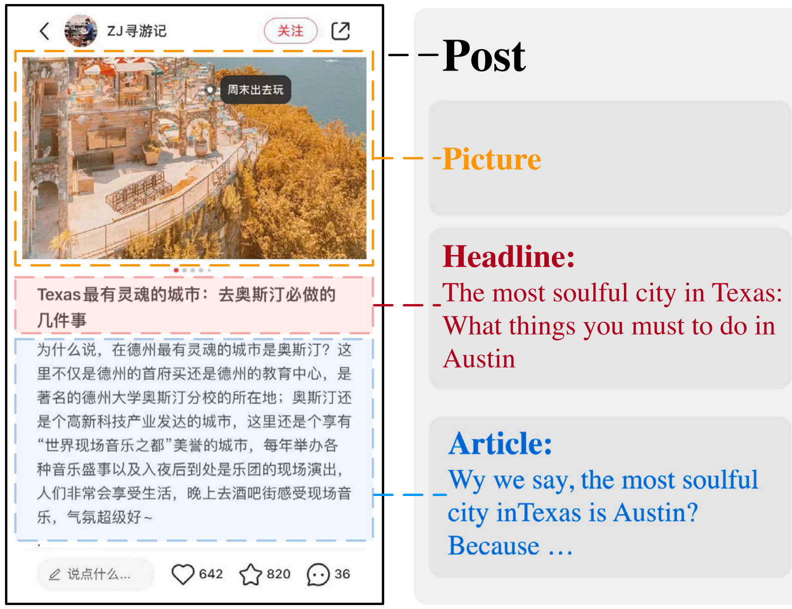

Fig. 1. A specific case of a popular post on Xiaohongshu.

writing (Shenassa and Minaei-Bidgoli, 2022). Most of methods focus on a single document (an article and a headline), and aim to extract and summarize the text's key points. Traditional headline generation task mainly focuses on news headlines, where faithfulness and briefness are the main concern. However, generating headlines on social media is more complicated, where the popularity of headline is also taken into consideration. To increase their headlines' popularity, social media users often follow trends (Wang et al., 2019), in order to enhance headlines' exposure on the media platform. Hence, headline generation should intuitively consider current trends on social media. Besides, some of the users have their unique and influential styles that attract a great deal of attention (Liu and Suh, 2017). Therefore, personal style is also essential for headline generation on social media. Considering these key factors, directly extending current methods of headline generation on social media may not achieve satisfactory performance.

However, it is challenging to include trends and personal styles in headline generation. First, it is difficult to figure out how trends and personal properties affect the generation process and popularity of headlines. To the best of our knowledge, there is no study exploring these issues on large-scale and real-world datasets. Furthermore, how to encode trends and personal styles is also a great problem. Trends and personal styles always change over time, which further complicates the encoding process.

To overcome these challenges, this paper collected more than 1 million posts from 42,447 users from the public data of Xiaohongshu. We conduct careful observations on the headlines of these posts. Results demonstrate that trends and personal styles indeed exist in the headlines and contribute significantly to posts' popularity. Inspired by these observations, we present *MEBART*, which combines **M**ultiple preference-**E**xtractors with **B**idirectional and **A**uto-**R**egressive **T**ransformers (BART) (Lewis et al., 2019), modeling trends and personal styles for headline generation on social media. Specifically, the preference-Extractors encode buzzwords on the platform and users' previous headlines, then output the corresponding preference encodings. T These encodings are combined with the embeddings of the article to generate the final headline. To improve the ability of preference-Extractors, we have designed a novel denoising-based pretraining strategy. In the pretraining stage, the target headline is first corrupted with token masking and replacing. *MEBART* then combines the corrupted headline, corresponding style encoding and trend encoding to reconstruct the original target headline.

Extensive experiments on the real-world dataset of Xiaohongshu demonstrate the effectiveness of our proposed model: *MEBART* significantly outperforms 3 SOTA baselines (average **9.9%** in term of ROUGE-L). We also conduct in-depth ablation studies to figure out how different components contribute to the performance of *MEBART*. Several cases are displayed to illustrate that *MEBART* advances in capturing trends and personal styles.

Overall, the contributions of this paper can be described as follows:

- We present a novel study: automatically generating popular headlines on social media; and investigate the problem from the perspectives of trends and personal styles.
- Comprehensive observations of over 1 million posts from Xiaohongshu illustrate the importance of trends and personal styles for creating popular headlines.
- Motivated by our observational insights, we propose *MEBART*, with a novel pretraining strategy, for headline generation. Experiment results on real-world datasets validate the effectiveness of our proposed method.

2. Preliminaries

For post z_i, x_i^{ar}, x_i^{se}, x_i^{te} and y_i are respectively the article, style text, trend text and headline. Time step t_x is the xth month-grained time step in increasing order. In other words, $t_x = t_{x-1} + 1$. T_{z_i} is the time step of z_i, demonstrating the month when the post is released. Y^{t_x} is all the headlines posted within time step t_x. $tf(w, t_x, V)$ is the token frequency of token w in vocabulary V for all the headlines within time step t_x. $tf(t_x, V)$ refers to all the token frequencies in V within time step t_x. $BW(t_x, V)$ is the buzzwords list of t_x in vocabulary V. W_{z_i} denotes the author(user) of z_i and Z_{u_a} denotes the posts list of user u_a.

3. Observation

In this section, we conduct in-depth analyses of headlines on social media platform. We mainly focus on two key factors for creating headlines: personal styles and trends. First, we introduce the dataset used for the analyses. Then, for each factor, we explore whether it is widespread in the headlines of posts and measure its contribution to posts' popularity.

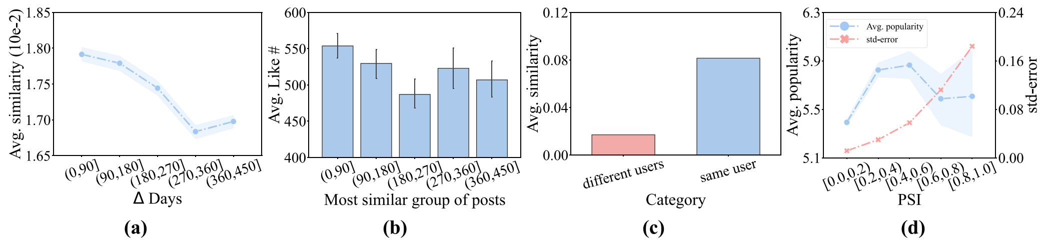

Fig. 2. Observations of the trends and personal styles of headlines. (a) shows the average similarities between the headlines from different seasons. (b) shows the average likes of headlines from different season groups. A headline is distributed to the season group the similarity between which is the highest among all.(c) shows the average similarities between the headlines of different users and the same users. (d) shows the average popularity of users with different extent of personal styles.

3.1. Dataset descriptions

The dataset for our experiments is described as follows: we sample posts on Xiaohongshu during a specific period of time from 2020 to 2022. The raw dataset contains 1695,219 posts and 42,447 users. Each post consists of basic information, including: (1) the article; (2) the headline; (3) received likes; (4) when the post is posted, namely time step; and (5) who creates the post, namely the user.

3.2. Analyses of trends

Existence of Trends

A trend is a change or development towards something new or different, which can be revealed in many aspects, including hot topic, buzzwords and prevalent sentence patterns. Take buzzwords for example, the Google Trends[3] has shown that lots of buzzwords appear on the Internet every day. However, it remains opaque that whether trends is widespread in social medias as well. Specifically, does a user take trends into consideration when he or she creates a headline? Therefore, we define the first research question: **Are users' headlines influenced by trends on social media? (Q1)**

To address this question, We first sampled 50,000 headlines from the last 10 days of 2021 as our study object, namely Y^{ob}. Then we sample 500 headlines for each of five different seasons. Each season is 90 days long and adjacent to its precedent. The first season is closest to Y^{ob}. Finally, we calculate the average text similarities between headlines from Y^{ob} and different seasons. In other words, we measure the similarity between headlines from the last 10 days of 2021 and from different seasons before the last 10 days of 2021.

We use *words overlap* as the metric of text similarity, which can be clarified as:

$$\mathbf{SIM}(y_i, y_j) = \frac{|set(y_i) \cap set(y_j)|}{|set(y_i) \cup set(y_j)|}, \tag{1}$$

where y_i and y_j denote two different headlines and $set(\cdot)$ is the set operation that removes duplicate words. The higher the SIM, the more similar the two headlines are.

The results are shown in Fig. 2(a): the average SIM reaches the maximum of 0.0179 in the most recent season and the minimum of 0.0168 in the fourth most recent season. In general, the SIM decreases as the difference of publishing time increases. However, when the difference of publishing time reaches a year, the SIM starts to rise back instead. The reason lies behind this may be the cyclical nature of trends (Robinson, 1975). We notice that the frequencies of some words, such as Christmas and Chinese New Year, are strongly associated with the time. Besides, although some words are not directly related to time, such as a festival or anniversary, we can still observe significant rise of frequencies of them. We think these words can reveal how would the users prefer to describe things, contemporary prevalent topics and other properties that are related to trends.

To sum up, based on the analysis above, we conclude that headlines are influenced by trends. In other words, **trends do exist in headlines on social medias**.

Trends Influence Popularity

The above analysis demonstrates the existence of trends on Xiaohongshu, which influence the creating of headlines for many users . However, it remains unknown that whether following trends help raise popularity of the posts. Therefore, we introduce the second research question: **What effect does the trends exert on the popularity of the posts? (Q2)**

To investigate this question, we continue to use the same Y^{ob}, separated seasons and metric of similarity above. Each headline in Y_{ob} are marked by the season in which the headlines are most similar to it. Then the headlines are divided into five groups according to their marks. Finally, We calculated the average number of likes for each group.

The results are shown in Fig. 2(b): the average likes reaches the maximum of 553.78 in the most recent season and the minimum of 486.93 in the third most recent season. In general, as the difference of publishing time increase, the number of likes decreases. But when the difference of publishing time reaches one year, the number of likes rises again. Since the difference of publishing time is roughly negative correlated with the extent of following trends as shown in the previous chapter, these results have shown the positive correlation of the extent of following trends and received likes.

To conclude, **a headline which follows trends has positive influence on the popularity of the post.**

3.3. Analyses of personal styles

Existence of Personal Styles

In the field of literature, many writers have their own styles of writing. For example, the style of Hemingway is recognized as abbreviated and simple while that of Jane Austen is recognized as fine and detailed. However, do personal styles exist in social medias as well? Social media platforms such as Xiaohongshu have abundant users and each of them may has his/her own style. Therefore, we introduce the third research question: **Do personal styles exist in creating headlines on social medias? (Q3)**

To address this question, we compare text similarities between the headlines from the same users and different users. Specifically, we first sample $1,000,000$ headline pairs, in which the two headlines are from the same user, and calculate the expectation of text similarity between headlines for all the pairs. Then we do the same for the headline pairs in which the two headlines are from different users. As in Section 3.2, we use *words overlap* as the metric of text similarity.

The results are shown in Fig. 2(c): the average SIM of the headline pairs from the same users is 0.081, much higher than that (0.017) of the headline pairs from different users. This demonstrates that headlines of

[3] https://trends.google.com/trends/

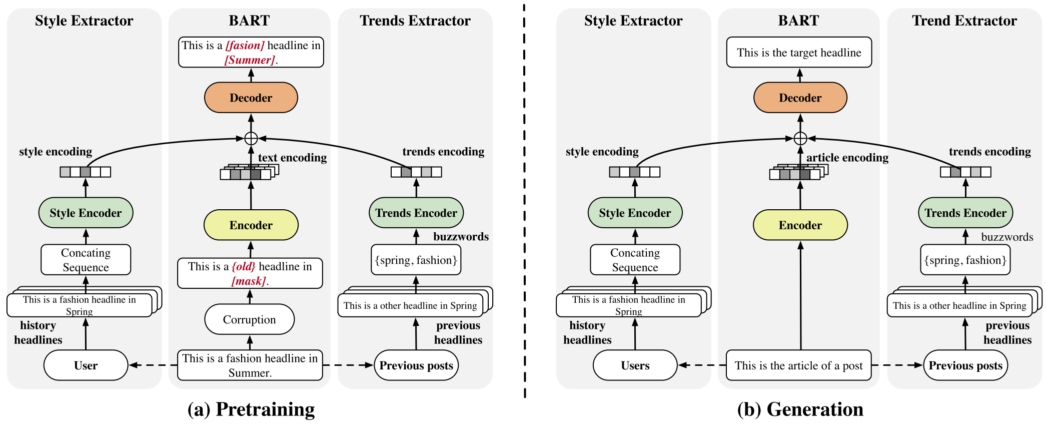

Fig. 3. The overview of *MEBART*. (a) displays the pipeline of *MEBART* during the pretraining stage, and (b) illustrates the pipeline of *MEBART* during the generating (including training and testing) stage.

the same users are more similar compared to the headlines of different users. To better understand this result, we observe several cases and find out a user is inclined to imitate his or her old headlines when making a new one in many aspects, including the typesetting patterns and choices of words.

In other words, **personal styles do exist in headlines on social medias.**

Personal Styles Influence Popularity

Although we are informed of the existence of personal styles, its relationship with post popularity remains unknown. Just like everyone has a different taste for books, readers may be attracted by headlines of different styles. Intuitively, users with apparent personal style may be more likely to be identified by the readers, thus drawing more attention. To investigate the relationship between personal styles and post popularity, we introduce the fourth research question: **What effect does the personal styles in headlines have on post popularity? (Q4)**

First, to quantify the personal styles, we introduce the *Personal Style Index* ($PSI \in [0, 1]$), which denotes the extent of personal style in the headlines of a user. Calculating PSI for user u_a is defined as:

$$\mathbf{PSI}(u_a) = \frac{\sum_{y_i \in Y_{u_a}} \sum_{y_j \in Y_{u_a}} \mathbf{SIM}(y_i, y_j)}{|Y_{u_a}|^2}, \tag{2}$$

where Y_{u_a} denotes the posted headlines of user u. Higher PSI demonstrates that the user has a more distinctive personal style. When PSI is equal to 1, it means that all the headlines of the user are exactly the same.

Then, to quantify the popularity of users, we introduce the *popularity index* (PI) as follows:

$$\mathbf{PI}(u_a) = \frac{\sum_{z_i \in Z_{u_a}} log(likes(z_i))}{|Z_{u_a}|}, \tag{3}$$

where Z_{u_a} denotes all the posts of user u_a and $likes(\cdot)$ denotes the number of likes of post z_i.

Finally, we divide the users into five groups according their PSI, and calculate PI for each group. The results are illustrated in Fig. 2(d): the PI of all users is 5.43, reaching the maximum of 5.86 when PSI is in the interval of $[0.4, 0.6)$ and the minimum of 5.39 when PSI is in the interval of $[0.0, 0.2)$. In general, as the PSI increases from 0 to 1, the average popularity of users first increases and then decreases. We also calculate the standard error of popularity for each interval shown in Fig. 2(d): the standard error increases from 0.011 to 0.184 as PSI becomes larger. These results suggest that posts with stronger personal styles are generally more popular compared to those without personal styles. However, if the personal style are too strong, the popularity may drop instead. The reason lies behind the drop may be that as the proportion of fixed words in the headlines becomes too large, there are few useful information for the readers in the headlines, which discourage them from clicking into the post.

To conclude, **personal styles within appropriate extent exert positive effect on the popularity of posts.**

4. Method

4.1. Overview of MEBART

The observations have demonstrated the existence of personal styles and trends, as well as their influences on popularity of posts on social medias. Inspired by these, we propose *MEBART* for generating personalized and trend-aware headlines.

In general, *MEBART* is a BART-based headline generator composed of four components, namely Encoder $f_{en}(\cdot)$, Decoder $f_{de}(\cdot)$, Style Extractor $f_{se}(\cdot)$ and Trend Extractor $f_{te}(\cdot)$. The architecture of *MEBART* is shown in Fig. 3. Like most of Transformer-based auto-regressors, firstly $f_{en}(\cdot)$ encodes the article into a sequence of hidden states $H_i^{ar} = (h_{i1}^{ar}, h_{i2}^{ar}, \ldots, h_{in_i}^{ar})$, then $f_{de}(\cdot)$ takes H_i^{ar} as input to generate a headline. Based on this framework, we introduce two kinds of preference-Extractors ($f_{se}(\cdot)$ and $f_{te}(\cdot)$) to model styles and trends respectively. The preference-Extractor takes representative text of the author's style or contemporary trends as input, and generate corresponding encoding as output. For an article x_i^{ar}, h_i^{se} and h_i^{te} respectively denote the **style encoding** and **trend encoding**. Since the qualities of h_i^{se} and h_i^{te} are crucial to the final generated headline, we propose a self-supervised learning task to pretrain $f_{se}(\cdot)$ and $f_{te}(\cdot)$. During generation, h_i^{se} and h_i^{te} are combined together to modify H_i^{ar}, guiding the model to generate personalized and trend-aware headlines. Details of how each component functions and the whole working pipeline of *MEBART* are described below.

4.2. Modeling personalized-styles and trends

Collecting Posted Titles and Buzzwords

The intention of the preference-Extractors is to model the "preferences" in styles and trends. As mentioned in the overview, for an article x_i^{ar}, the preference-Extractors require representative text (**style text** x_i^{se} and **trend text** x_i^{te}) as input. The qualities of x_i^{se} and x_i^{te} largely determine the qualities of h_i^{se} and h_i^{te}, and indirectly influence the model's ability to capture styles and trends. However, it is challenging to form x_i^{se} and x_i^{te} that can truly **speak for** styles and trends, since they are quite complex and integrated concepts. In other words, we can hardly define styles and trends from plain intuition. Therefore, we carefully design the strategies for constructing x_i^{se} and x_i^{te}.

For personal styles, we have be informed by the observation that many users have their own styles in creating headlines, which help them quickly draw the attention of their "old fans". Therefore, we use the posted headlines of the same user to form x_i^{se}, as a reference for the new headline to be generated. For post z_i and its author u_a, x_i^{se} is formatted by concating the headlines whose corresponding post z_j satisfies $z_j \in Z_{u_a}$ and $T_{z_j} < T_{z_i}$ in decreasing order of time step.

For trends, the observation has shown that the similarity between headlines increases as the difference of their publishing time decreases, which supports the existence of trends. The observation also demonstrates that following trends helps exert positive effect upon the popularity of posts. However, different from personal styles, directly concating all the headlines within t_x to form trend text for all the article within t_x is unreasonable. Considering the amount of Y^{t_x}, this approach may introduce abundance of duplicated and noisy text. Within limited training time, it is too hard for the model to learn. To effectively represent the trends, we use buzzwords (Nakajima et al., 2012) instead, which are prevalent words during a certain period. The first way of selecting buzzwords that comes to mind may be sorting words by their token frequency (tf). However, it is not appropriate to simply define the buzzwords as words with **absolutely-high** tf in the moment. This kind of definition may ignore some important rising trends, which will be popular in the near future. For example, a new APPLE product, a new record of Rihanna, or news that has just been reported, etc. Therefore, we make a trade-off. That is, when collecting the buzzwords, we also pay attention to those words with **relatively-high** tf compared to the past.

To be specific, we first collect headlines in the train dataset and use them to generate a headline vocabulary V. Words are filtered with minimum tf (tf_{min}) and maximum tf (tf_{max}) . Then we calculate $tf(t_x, V)$ for every time step t_x. In real-world, users can only take the headlines in the past as references for creating a new headline. Therefore, to generate buzzwords list for t_x, we can only use the information of tf before t_x. In other words, $BW(t_x)$ should be produced with $tf(t_y, V)$ where $t_y < t_x$ only. So $BW(t_0, V)$ is actually empty. Finally, tokens with different time-grained relatively-high tf and absolutely-high tf are selected in balance. All the articles within t_x share the same $BW(t_x, V)$, which will be concated to form the same trend text during generation. Detailed algorithm of generating $BW(t_x, V)$ can be formulated as in Algorithm 1.

Preference-Extractor

Inspired by Riley et al. (2020), we propose the preference-Extractor, which is a modification based on the encoder of BART. To be specific, the preference-Extractor is composed of two part - a transformer-based encoder (Vaswani et al., 2017) and a pooling layer. The former part is the same as BART's encoder;the latter part is a mean-pooling layer, attached behind the former part. The encoding sequence of the encoder is turned into a single length-fixed encoding(vector) after pooling. The dimensionality of the encoding is set to d, matched with that of the encoder's hidden states.

Parameters are not shared between preference-Extractors, except the token-embedding layer. For x_i^{ar}, $f_{se}(\cdot)$ takes style text x_i^{se} as input and generates style encoding h_i^{se} as output, while $f_{te}(\cdot)$ takes trend text x_i^{te} as input and generates trend encoding h_i^{te} as output. These can be clarified as:

$$h_i^{se} = f_{se}(x_i^{se}) \tag{4}$$

and

$$h_i^{te} = f_{te}(x_i^{te}) \tag{5}$$

Algorithm 1 Generate Buzzwords

Require: Time step: t_x; Headlines list: $\{Y^{t_0}, Y^{t_1}, Y^{t_2} ...\}$;Headline vocabulary: V;
Ensure: Buzzwords list $BW(t_x, V)$ for time step t_x
  if $t_x \geq 2$ **then**
    for $w \in V$ **do**
      $R(w) \leftarrow tf(w, t_{x-1}, V)/tf(w, t_{x-2}, V)$;
    end for
    Add the 128 tokens with highest $R(w)$ to $BW(t_x, V)$
  end if
  if $t_x \geq 6$ **then**
    for $w \in V$ **do**
      $R(w) \leftarrow tf(w, (t_{x-1 \ to \ x-3}), V)/tf(w, (t_{x-4 \ to \ x-6}), V)$;
    end for
    Add the 64 tokens with highest $R(w)$ to $BW(t_x, V)$
  end if
  if $t_x \geq 12$ **then**
    for $w \in V$ **do**
      $R(w) \leftarrow tf(w, (t_{x-1 \ to \ x-6}), V)/tf(w, (t_{x-7 \ to \ x-12}), V)$;
    end for
    Add the 32 tokens with highest $R(w)$ to $BW(t_x, V)$
  end if
  if $t_x \geq 1$ **then**
    $V' \leftarrow V$ sorted in descending order of $tf(w, t_{x-1}, V)$
    for $w \in V'$ **do**
      Add w to $BW(t_x, V)$
      if $sizeof(BW(t_x, V)) = 512$ **then**
        break
      end if
    end for
  end if

where the h_i^{se} and h_i^{te} are the results after mean-pooling the last hidden states of the encoder part of the $f_{se}(\cdot)$ and $f_{te}(\cdot)$ respectively. The dimension of them is d.

Pretraining strategy

As mentioned before, the intention of preference-Extractors is to capture the **preference** in styles and trends, eventually helping the model to generate personalized and trend-aware headlines. Therefore, we design a Masked LM task (Devlin et al., 2018) to improve preference-Extractors' ability. For y_i and corruption function $\phi(\cdot)$, the corrupted headline is $\widetilde{y_i} = \phi(y_i)$. We use the same setting of $\phi(\cdot)$ as (Devlin et al., 2018) does, which includes masking and replacing tokens. $\widetilde{y_i}$ is then used to predict the original y_i. To be specific, $\widetilde{y_i}$ is first fed into $f_{en}(\cdot)$ to produce a sequence of hidden state $H_i^{ti} = (h_{i1}^{ti}, h_{i2}^{ti}, \dots, h_{in_i}^{ti})$. Then each hidden state in H_i^{ti} are added up with h_i^{se} and h_i^{te}, producing the new hidden state sequence S_i^{ti}. Finally, $f_{de}(\cdot)$ uses S_i^{ti} to generate the prediction $\widehat{y_i}$. These can be formally expressed as:

$$\begin{aligned} H_i^{ti} &= f_{en}(\widetilde{y_i}) \\ H_i^{se} &= (h_i^{se}, h_i^{se}, \dots, h_i^{se}) \\ H_i^{te} &= (h_i^{te}, h_i^{te}, \dots, h_i^{te}) \\ S_i^{ti} &= H_i^{ti} + H_i^{se} + H_i^{te} \\ \widehat{y_i} &= f_{de}(S_i^{ti}) \end{aligned} \tag{6}$$

where H_i^{ti}, H_i^{se} and H_i^{te} has the same length n_i. S_i^{ti} is the modified hidden state sequence.

The target of pretraining is to reconstruct the original headline under the "direction" of personal styles and buzzwords. The model only predicts masked tokens during pretraining, since other tokens are not corrupted. Therefore, the loss function of pretraining stage can be formulated as follows:

$$L_{pre} = -\frac{1}{M_i} \sum_{m=1}^{M_i} \sum_{c=1}^{C} y_i^{mc} log(\hat{y}_i^{mc}) \tag{7}$$

where M_i is the number of masked tokens, C is the number of classes (possible tokens). $\hat{y}_i^{mc}$ is the predicted probability of class c for the mth masked token. y_i^{mc} is the real label of class c for the mth masked token.

Fig. 3(a) illustrates the process of pretraining. During pretraining, $f_{de}(\cdot)$ and $f_{en}(\cdot)$ will be frozen, including the token-embedding layer shared by all the four components of *MEBART*.

4.3. Headlines generation

Training procedure

During training, *MEBART* combines x_i^{ar}, x_i^{se} and x_i^{te} together to predict tokens $y_i = (y_i^1, y_i^2, \dots, y_i^{|y_i|})$. x_i^{ar} is fed into $f_{en}(\cdot)$ to produce a sequence of hidden state $H_i^{ar} = (h_{i1}^{ar}, h_{i2}^{ar}, \dots, h_{in_i}^{ar})$. Then each hidden state in H_i^{ar} are added up with h_i^{se} and h_i^{te}, which are the encodings of x_i^{se} and x_i^{te}, producing the new hidden state sequence S_i^{ar}. These can be formulated as follows:

$$\begin{aligned} H_i^{ar} &= f_{en}(x_i^{ar}) \\ H_i^{se} &= (h_i^{se}, h_i^{se}, \dots, h_i^{se}) \\ H_i^{te} &= (h_i^{te}, h_i^{te}, \dots, h_i^{te}) \\ S_i^{ar} &= H_i^{ar} + H_i^{se} + H_i^{te} \end{aligned} \tag{8}$$

where H_i^{ar}, H_i^{se} and H_i^{te} has the same length n_i. S_i^{ar} is the modified hidden state sequence.

The target in the training stage is to reconstruct the original headline with the help of personal styles and trends. Therefore, we define the loss function as:

$$L_{tra} = -\frac{1}{|y_i|} \sum_{q=1}^{|y_i|} log_{p_\theta}(y_i^q | y_i^{<q}, S_i^{ar}) \tag{9}$$

where S_i^{ar} is the modified hidden state sequence. y_i^q is the qth token of y_i.

Fig. 3(b) illustrates the process of training. All the parameters of *MEBART* are trainable in training.

Generation Deployment

During generation, we use Beam Search with a beam size of 4 to generate headlines. Beam Search is a searching technique that employs a length-normalization procedure and uses a coverage penalty, encouraging generation of an output sentence that is most likely to cover all the words in the source sentence. Wu et al. (2016) The maximum length of generated headlines is 20 (not including $< SEP >$), same as the limit for headlines on Xiaohongshu. Given that all the articles within the same time step t_x share the same style text, our model can be conveniently put into usage. For social media platforms, they can update the buzzwords regularly(such as once a week) and combine user's posted headlines and buzzwords to given several candidates of personalized and trend-aware headlines for a new post.

5. Experiments

5.1. Experiment settings

Datasets preprocessing. The raw dataset is described in 3.1. We preprocess the raw dataset for our experiments. Since we focus on generating **popular** headlines, we filter those posts with less than 500 likes. We also filter posts whose article or headline is empty, since they lacks input or output for the headline generation task. As described in 4.2, posted headlines are required to form the style text, so we filter posts whose user has no former posts, ensuring that every article has non-empty style text. Since the styles and trends in our experiments are time-sensitive, posts in the future should be "invisible" to *MEBART* during training. Therefore, we use the posts in 2021 for training and those in 2022 for validation and test. The preprocessed dataset contains 248,311 posts in total. The train dataset consists of 150,770 posts, all of which are in posted 2021. The validation and test datasets are randomly splitted from all the posts in 2022, consist of 9780 and 87,761 posts respectively.

Baselines. We compare our proposed model, *MEBART*, with a variety of advanced baselines. They are described below:

- BART (Lewis et al., 2019) is a denoising autoencoder for pretraining sequence-to-sequence models.
- PEGASUS (Zhang et al., 2020) is an abstractive summmarization sequence-to-sequence model pretrained with extracted gap-sentences.
- BRIO (Liu et al., 2022) proposed a training object that encourages coordination of probabilities and qualities among non-reference candidates generated by abstractive summarization models.

Evaluation metric. To evaluate the performance of each model, we use the metric called Recall-Oriented Understudy for Gisting Evaluation (ROUGE), which counts the number of overlapping units between the computer-generated summary to be evaluated and the ideal summary created by humans Lin (2004).

Implementation details. During building the headline vocabulary V, we set $tf_{min} = 10$ and $tf_{max} = 0.01$. That is, tokens that appear less than 10 times or tokens with more than 0.01 tf are not included in V. The maximum input sequence lengths of $f_{en}(\cdot)$, $f_{se}(\cdot)$ and $f_{te}(\cdot)$ are the same, namely 512. The random seed is set to 3407 for all our experiments. We initialize the weights of our model *MEBART* with those of a pretrained BART model (bart-base-chinese) (Shao et al., 2021). In detail, we initialize the style extractor $f_{se}(\cdot)$, trend extractor $f_{te}(\cdot)$ and encoder $f_{en}(\cdot)$ from the pretrained encoder of the BART, but the weights are not tied during training. We initialize the decoder $f_{de}(\cdot)$ from the pretrained decoder of the same BART above. The total size of *MEBART* is about 771M.

For pertraining and training, the batch size is 64 and the updating step is 4 batches. We use the Adam optimizer (Kingma and Ba, 2014) with learning rate scheduling:

$$lr = 2 \times 10^{-3} min(step^{-0.5}, step \cdot warmup^{-1.5}) \tag{10}$$

where lr is the learning rate, $step$ is the number of updating steps, $warmup$ denotes the warmup steps, which is set to 100.

To decide the appropriate number of epochs for pretraining, we pretrain *MEBART* for 5 epochs and evaluate its performance on the validation dataset at the end of each epoch. We then choose the pretrained model which achieves highest accuracy on the pretraining task (masked tokens prediction). The chosen pretrained model is then finetuned.

For training, to fairly measure each model's performance, we finetune all the models for 5 epochs and evaluate their performance on the validation dataset every 64 updating steps in terms of ROUGE F1 scores. For each model, we choose its parameters with the best ROUGE-L F1 score on the validation dataset, then evaluate its performance on the test dataset, reporting ROUGE-1/2/L F1 scores.

5.2. Results

We present the performance of all the models on headline generation task on the test dataset in Table 1. Overall, *MEBART* achieves the best performance, brings average increase of 9.8%, 12.3% and 9.9% in terms of ROUGE-1, ROUGE-2 and ROUGE-L compared with the baselines. But how could we relate the ROUGE scores with *MEBART* 's ability of modeling the styles and trends? To understand this, let us think about the process of creating a new headline from the perspective of the user. The observation has demonstrated significant similarity

Table 1
Experimental results on the test dataset of headline generation task. R-1/2/L are the ROUGE-1/2/L F1 scores.

Model	R-1	R-2	R-L
BART	28.33	15.19	25.09
PEGASUS	27.31	14.23	24.21
BRIO	28.79	15.32	25.32
MEBART	**30.88**	**16.73**	**27.33**

Table 2
Ablation study on *MEBART*. "w/o" means *MEBART* without a certain component. R-1/2/L are the ROUGE-1/2/L F1 scores.

Model	R-1	R-2	R-L
w/o Style Extractor	27.52	14.5	24.28
w/o Trend Extractor	30.21	15.94	26.68
MEBART	**30.88**	**16.73**	**27.33**

Table 3
MEBART pretrained with different epochs. R-1/2/L are the ROUGE-1/2/L F1 scores.

Epochs	R-1	R-2	R-L
0	29.57	15.39	26.04
1	**31.12**	**16.84**	**27.53**
2	30.88	16.73	27.33
3	30.78	16.47	27.24

between headlines of the same user, supporting the existence of personal styles. In other words, a user probably tends to create a new headline that is similar to the his/her old ones. Besides, the posts in our experiments are all popular posts, which means their headlines have great chance of containing buzzwords. Therefore, informed of the old headlines(personal styles) and the buzzwords(trends), *MEBART* is able to more accurately predict the new headline.

The performance of PEGASUS is worse than BART. We think this may result from the pretraining task of PEGASUS-removing/masking important sentences from the input document and generating them as one output sequence from the remaining sentences, similar to an extractive summary. PEGASUS thus tends to generate the headline by concating important tokens from the input document (article). However, unlike news, the article and its headline on social medias are not so tightly associated, which means the way in which PEGASUS generate headlines may not apply under this circumstance. The performance of BRIO is better than BART, since it is able to assign higher estimated probability to the better candidate summary during inference. In conclusion, due to the ability of modeling styles and trends, *MEBART* achieves the best performance compared to other baselines.

5.3. In-depth analysis of *MEBART*

The effect of different components. *MEBART* consists of two preference-Extractors, Transformer based encoder and decoder. To investigate whether the two preference-Extractors, namely style extractor and trend extractor, actually work and how they influence the performance of *MEBART*, we conduct ablation studies by removing one of them at a time, and evaluate the performances of these variants of *MEBART* in Table 2

For every variant, we use the same setup of pretraining and training as *MEBART*. Style extractor $f_{se}(\cdot)$ and trend extractor $f_{te}(\cdot)$ are designed to capture the styles and trends respectively. We compare the performance of *MEBART* without one of these components. Removing style extractor leads to a decline of 10.9%, 13.3% and 11.2% in terms of ROUGE-1, ROUGE-2 and ROUGE-l. Removing trend extractor leads to a decline of 2.2%, 4.7% and 2.4% in terms of ROUGE-1, ROUGE-2 and ROUGE-l. We notice that the variant with only $f_{te}(\cdot)$ perform worse than BART. However, using $f_{se}(\cdot)$ and $f_{te}(\cdot)$ together does improve the proposed model compared to using $f_{se}(\cdot)$ only. We think only using $f_{te}(\cdot)$ may overemphasize the importance of buzzwords, thus misleading the proposed model. But when combined with $f_{se}(\cdot)$, the misleading is corrected under the supervision of personal styles. In other words, the proposed model pays more attention to the trends that truly cooperate with personal styles.

To summarize, according to the above discussions: (1) combining style extractor and trend extractor improves the proposed model to a large extent (2) only using style extractor can also improve the proposed model to a smaller extent, but using only trend extractor has the negative effect.

The effect of pretraining. Preference-extractors in *MEBART* are pretrained with a Masked LM task. To investigate whether pretraining actually improves the model, we finetune *MEBART* pretrained with different epochs and evaluate the performances of these versions of *MEBART* in Table 3.

For every version, we use the same setup of training as stated in . $MEBART_n$ denotes different version of proposed model pretrained for n epochs. The results show that $MEBART_1$ performs the best, bringing increase of 5.2%, 9.4% and 5.7% in terms of ROUGE-1, ROUGE-2 and ROUGE-L compared with $MEBART_0$. This demonstrates considerable positive effect of pretraining for the proposed model. We also notice that although $MEBART_2$ achieves highest accuracy for the pretraining task (masked token prediction), its performance on downstream task(headline generation) is slightly worse than $MEBART_1$. We think this is because that the input text for the encoder $f_{en}(\cdot)$ is different during pretraining and finetuning. Therefore, the performance on pretraining task cannot fully reflect the ability of preference-extractors. To summarize, the above discussions demonstrate our pretraining task benefits the performance of *MEBART*.

5.4. Case study

To get a more direct look of *MEBART*'s ability of modeling styles, we compare several headlines for the same article respectively created by the user, *MEBART* and BART, presented in Table 4. (a_1, a_2, a_3) and (b_1, b_2, b_3) are listed in the order of time. Cases in (a_1, a_2, a_3) show that *MEBART* is able to imitate user's style, such as the same pattern "motd |". In b_3, *MEBART* is able to generate the pattern "ootd |", which can be recognized as the personal style of USER B, even when the user himself/herself actually does not use "ootd |" for the article.

Cases in Table 5 show *MEBART*'s ability of modeling trends. In the first case, *MEBART* is able to generate the same underlined token which means "spring and summer" as in the human-created headline. In the second case, informed of trend text, *MEBART* is able to generate the underlined token which means "early spring" even when it is not in the human-created headline.

To summarize, these examples demonstrate that *MEBART* has the ability to model personal styles and trends.

6. Related work

Headline generation is a popular field of research, which can be regarded as a branch of summarization. Transformer (Vaswani et al., 2017) based methods for headline generation have shown great potential these years (Lewis et al., 2019; Zhang et al., 2020; Hasan et al., 2021b,a; Liu et al., 2022; Xu et al., 2022). The applications of headline generation varies in different fields, including news (Ao et al., 2021; Zheng et al., 2021), online ads (Kanungo et al., 2022), paper writing (Shenassa and Minaei-Bidgoli, 2022; Xu et al., 2022) and content community (Zhang et al., 2022). Recently, there are also works that starting to focus on generating stylistic headlines Zhang et al. (2018), Jin et al. (2020), Cao and Wang (2021). However, these methods either are limited to several specific styles or demand a text

Table 4
Example headlines generated by *MEBART* and BART. REF denotes the headlines created by the user. USER A prefers the pattern “motd | ” in his/her headline, while USER B prefers “ootd”.

USER	CASE	REF	MEBART	BART
A	a_1	motd | 今天是chanel girl	motd | 秋冬氛围感妆容	今天是小香味的一天
	a_2	motd | 冬日清冷易碎感	motd | 清冷易碎感妆容	清冷易碎感碎钻妆容
	a_3	motd | 浅画一个春天	motd | 绿野仙踪	绿野仙踪 | 春天来了
B	b_1	ootd早春甜蜜撞色混搭分	ootd | 春夏撞色穿搭	早春穿搭 | 今年春夏季节绿色系和粉色系都
	b_2	ootd 早春草莓泡泡糖色系好嫩哟	ootd | 粉粉嫩嫩的休闲风穿搭	早春穿搭 | 芭比粉真的是千禧年y2k辣妹艺
	b_3	《我喜欢把衣服穿出花样》～	ootd | 一周穿搭不重样	早春穿搭 | 美式复古风

Table 5
Example headlines generated by *MEBART* and BART. REF denotes the headlines created by the user. Time-related tokens are underlined.

TIME	REF	MEBART	BART
03/2022 (spring)	<u>春夏</u> (spring and summer) 小裙子购物分享我可真是爱惨啦！！！	<u>春夏</u> (spring and summer) 小裙子合集来啦	小裙子合集来啦！快来抄作业！ (none)
01/2022 (spring)	165/44 | 全套链接！甜辣学院开衫&毛衣&袜 (none)	ootd | <u>早春</u> (spring) 毛衣开衫外套穿搭合集	ootd | 毛衣开衫外套穿搭韩系学院风辣妹 (none)

corpus for the target style. Therefore, considering the numerous users (everyone has his/her own style) and the relatively small corpus for each user (his/her posted posts), it is not appropriate to directly apply these methods to generate headlines for posts on social media.

7. Conclusion

This paper present a novel study: automating generating popular headlines on social medias; and investigate the problem from the perspectives of trends and personal styles. To figure out how trends and personal styles affect the generation process and popularity of headlines, this paper collected more than 1 million posts from 42,447 thousand from public data of Xiaohongshu and conduct careful observations on the headlines of these posts. Results suggest that trends and personal styles are indeed widespread in the headlines of posts and have significant impacts on posts’ popularity. Therefore, we present *MEBART*. It is a BART-based method, with two special preference-Extractors to capture trends and personal styles. We conduct extensive experiments on Xiaohongshu datasets. *MEBART* achieves the best performance on the test dataset, bringing average increase of 9.9% in terms of ROUGE-L compared with the baselines. Meanwhile, ablation studies demonstrate that modeling trends and personal styles indeed prompt the model’s performance. Besides, the results also demonstrate the proposed pretraining strategy is powerful. To further study the advancement of *MEBART*, we present several cases. In summary, these results demonstrate that *MEBART* indeed is an effective, trends-aware, personalized headline generation method on social medias.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Min Yu and Xiaoqi Tang are currently employed by Huayun Information and Technology Co. Ltd.

References

Ao, X., Wang, X., Luo, L., Qiao, Y., He, Q., Xie, X., 2021. Pens: A dataset and generic framework for personalized news headline generation. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 82–92.

Bradshaw, A.S., Treise, D., Shelton, S.S., Cretul, M., Raisa, A., Bajalia, A., Peek, D., 2020. Propagandizing anti-vaccination: Analysis of Vaccines Revealed documentary series. Vaccine 38 (8), 2058–2069.

Cao, S., Wang, L., 2021. Inference time style control for summarization. arXiv preprint arXiv:2104.01724.

De Vries, L., Gensler, S., Leeflang, P.S., 2012. Popularity of brand posts on brand fan pages: An investigation of the effects of social media marketing. J. Interact. Market. 26 (2), 83–91.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Gardner, J., Lehnert, K., 2016. What’s new about new media? How multi-channel networks work with content creators. Bus. Horiz. 59 (3), 293–302.

Hasan, T., Bhattacharjee, A., Ahmad, W.U., Li, Y.-F., Kang, Y.-B., Shahriyar, R., 2021a. CrossSum: Beyond english-centric cross-lingual abstractive text summarization for 1500+ language pairs. arXiv preprint arXiv:2112.08804.

Hasan, T., Bhattacharjee, A., Islam, M.S., Samin, K., Li, Y.-F., Kang, Y.-B., Rahman, M.S., Shahriyar, R., 2021b. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. arXiv preprint arXiv:2106.13822.

Jin, D., Jin, Z., Zhou, J.T., Orii, L., Szolovits, P., 2020. Hooks in the headline: Learning to generate headlines with controlled styles. arXiv preprint arXiv:2004.01980.

Kanungo, Y.S., Das, G., Negi, S., 2022. COBART: Controlled, optimized, bidirectional and auto-regressive transformer for ad headline generation. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 3127–3136.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Lakkaraju, H., McAuley, J., Leskovec, J., 2013. What’s in a name? understanding the interplay between titles, content, and communities in social media. In: Proceedings of the International AAAI Conference on Web and Social Media, vol. 7. pp. 311–320.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L., 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461.

Lin, C.-Y., 2004. Rouge: A package for automatic evaluation of summaries. In: Text Summarization Branches Out. pp. 74–81.

Liu, Y., Liu, P., Radev, D., Neubig, G., 2022. BRIO: Bringing order to abstractive summarization. arXiv preprint arXiv:2203.16804.

Liu, R., Suh, A., 2017. Self-branding on social media: An analysis of style bloggers on Instagram. Procedia Comput. Sci. 124, 12–20.

McMillan, S.J., Morrison, M., 2006. Coming of age with the internet: A qualitative exploration of how the internet has become an integral part of young people’s lives. New Media Soc. 8 (1), 73–95.

Meij, E., Weerkamp, W., De Rijke, M., 2012. Adding semantics to microblog posts. In: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining. pp. 563–572.

Morrison, M.A., Cheong, H.J., McMillan, S.J., 2013. Posting, lurking, and networking: behaviors and characteristics of consumers in the context of user-generated content. J. Interact. Advert. 13 (2), 97–108.

Nakajima, S., Zhang, J., Inagaki, Y., Nakamoto, R., 2012. Early detection of buzzwords based on large-scale time-series analysis of blog entries. In: Proceedings of the 23rd ACM Conference on Hypertext and Social Media. pp. 275–284.

Riley, P., Constant, N., Guo, M., Kumar, G., Uthus, D., Parekh, Z., 2020. TextSETTR: Few-shot text style extraction and tunable targeted restyling. arXiv preprint arXiv:2010.03802.

Robinson, D.E., 1975. Style changes: Cyclical, inexorable, and foreseeable. Harv. Bus. Rev. 53 (6), 121–131.

Saravanakumar, M., SuganthaLakshmi, T., 2012. Social media marketing. Life Sci. J. 9 (4), 4444–4451.

Shao, Y., Geng, Z., Liu, Y., Dai, J., Yang, F., Zhe, L., Bao, H., Qiu, X., 2021. CPT: A pre-trained unbalanced transformer for both Chinese language understanding and generation. arXiv preprint arXiv:2109.05729.

Shenassa, M.E., Minaei-Bidgoli, B., 2022. ElmNet: a benchmark dataset for generating headlines from Persian papers. Multimedia Tools Appl. 81 (2), 1853–1866.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 30.

Wang, X., Zhang, B., Chang, F., 2019. Hot topic community discovery on cross social networks. Future Internet 11 (3), 60.

Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al., 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144.

Xu, S., Zhang, X., Wu, Y., Wei, F., 2022. Sequence level contrastive learning for text summarization. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36. pp. 11556–11565.

Zagovora, O., Weller, K., Janosov, M., Wagner, C., Peters, I., 2018. What increases (social) media attention: Research impact, author prominence or title attractiveness? In: STI 2018 Conference Proceedings. Centre for Science and Technology Studies (CWTS), pp. 1182–1190.

Zhang, R., Guo, J., Fan, Y., Lan, Y., Xu, J., Cao, H., Cheng, X., 2018. Question headline generation for news articles. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. pp. 617–626.

Zhang, F., Liu, J., Wan, Y., Yu, X., Liu, X., Keung, J., 2022. Diverse title generation for stack overflow posts with multiple sampling enhanced transformer. arXiv preprint arXiv:2208.11523.

Zhang, J., Zhao, Y., Saleh, M., Liu, P., 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In: International Conference on Machine Learning. PMLR, pp. 11328–11339.

Zheng, X., Sun, A., Muthuswamy, K., 2021. Tweet-aware news summarization with dual-attention mechanism. In: Companion Proceedings of the Web Conference 2021. pp. 473–480.