Computational Linguistics and Chinese Language Processing
Vol. 24, No. 1, June 2019, pp. 15-38

Discovering the Latent Writing Style from Articles: A Contextualized Feature Extraction Approach

Yen-Hao Huang[*], Ting-Wei Liu[*], Ssu-Rui Lee[*], Ya-Wen Yu[*], Wan-Hsuan Lee[*], Fernando Henrique Calderon Alvarado[*] and Yi-Shin Chen[*]

Abstract

With the growth of the Internet, the ready accessibility and generation of online information has created the issue of determining how accurate or truthful that information is. The rapid speed of information generation makes the manual filter approach impossible; hence, there is a desire for mechanisms to automatically recognize and filter unreliable data. This research aimed to create a method for distinguishing vendor-sponsored reviews from customer product reviews using real-world online forum datasets. However, the lack of labelled sponsored reviews makes end-to-end training difficult; many existing approaches rely on lexicon-based features that may be easily manipulated by replacing word usages. To avoid this word manipulation, we derived a graph-based method for extracting latent writing style patterns. Thus, this work proposes a Contextualized Affect Representation for Implicit Style Recognition framework, namely CARISR. Transfer learning architecture was also adapted to improve the model's learning process with weakly labeled data. The proposed approach demonstrated the ability to recognize sponsored reviews through comprehensive experiments using the limited available data with 70% accuracy.

Keywords: Reliability, Transfer Learning, Writing Style, Text Classification, Natural Language Processing.

[*] Department of Computer Science, National Tsing Hua University
E-mail: yenhao0218@gmail.com

1. Introduction

With the popularization of the Internet and communication devices, information can be sent more quickly and widely than ever before. However, technological advances have also made it difficult to avoid incorrect information. Sponsored reviews, which have recently become a popular marketing strategy in online forums, can provide incorrect information. The intention of these articles is to give their consumers a positive impression of the product. Some advertisement companies have even begun to use sponsored reviews as a new method of promoting their commodities. Such sponsored reviews usually only provide positive information about a product. Thus, these reviews may hide the disadvantages of a product and potentially mislead consumers into making an unbeneficial purchase.

As unreliable data may contain incomplete or incorrect information, it is important to avoid them. Most of the filtering approaches on online social platforms rely on mutual reviewing from users or human-designed rules. However, no matter which approach is used, automatic filtering is still limited due to the various methods of writing sponsored reviews and how quickly information is generated. Consequently, a system to automatically identify these kinds of information has become an important issue in the information reliability research field.

In this work, we focus on recognizing the information reliability of review articles on online web platforms. Review articles are widely consumed by readers in order for them to purchase the best products. General filtering methods fail to address two main difficulties. First, current filters are easily fooled if the method only considers word-based characteristics; writers can simply avoid specific words/phrases to pass the filtering check. Second, there is a lack of defined and labeled sponsored review article data for testing reliability problems. It is difficult enough to manually collect these articles, let alone to create rules for automatically gathering them, because these articles are written by experienced writers.

To address the first issue of keywords bias, this research focused on extracting the latent writing style of review articles to avoid specific word biases found in word-level methods. The presented research proposes a Contextualized Affect Representation for Implicit Style Recognition (CARISR) method to recognize the writing styles of various reviews. The proposed CARISR consists of an unsupervised approach for generating stylistic word patterns, which condenses patterns into distributed matrix representations, and a learning-based model. Sections 4 and 5 describe the details of the stylistic patterns and model, respectively.

The biggest difference between the general methods and CARISR is that the latter defines two specific word groups, stylistic skeleton words (CW) and stylistic content words (SW), to capture the writing style information. A set of stylistic word patterns are extracted based on the constructive relationship of different stylistic skeleton words and content words in the sentence. By adopting stylistic word patterns, the experiment results show that CARISR is more robust compared to the word-based approaches, including neural network methods. In other words, the contextualized effect representation model is less susceptible to changes to specific words. Consequently, CARISR has a better ability to deal with the first challenge, that is, to detect the implicit word usages of advertisement writers.

For the second difficulty, the lack of labeled data, we defined our recognized targets as sponsored reviews (業配文), trial product reviews (產品試用文), and self-purchased product reviews (自購心得文). Since it is rare for sponsored reviews to actually be labelled as such, we introduced a similar class that is more easily obtained, called official advertisements (廣告), as the weak label concept for model pre-training. The transfer learning approach can then be applied to the target label of sponsored review.

This work proposes that the purpose of the sponsored review is more similar to official advertisements than self-purchased product reviews. This similarity allows for transfer learning to be adopted in our work. After preliminary training leveraging a large number of advertisements, the model should have the ability to classify the implicit writing style of advertisements. Further, we manually collect small amounts of sponsored review for transfer learning and fine-tune. The proposed model achieves around 70 percent accuracy and shows better robustness than the compared models, which demonstrates that our framework works successfully, even with the scarce sponsored review resources.

To shortly summarize this research, we highlight the following contributions:

- To quantify the problem of review articles' reliability, we defined different levels of reviews and collect the corresponding dataset for the training model.
- To prevent our model from being defrauded by intentional word selection, our model recognizes reliability based on the implicit writing style instead of word-level features.
- To capture the implicit writing style, we first applied graph-based pattern extraction to the review articles. Then, we designed the embedding strategy of contextual stylistic patterns for the convolutional neural network model.
- To overcome the insufficient quantity problem, we combined the weak label concept and the transfer learning approach to stabilize the learning process and improve the performance and robustness of our model.

2. Related Work

2.1 Information Reliability

Information reliability research aims to distinguish whether the given information is reliable or not. Most of the information reliability research could be consider as credibility analysis on news. The main difficulty of credibility analysis is how to find the effective features to identifying the news is reliable or not. To address the problems, the researchers attempt to extract different features, which could be categorized as the propagation-based, knowledge-based and content-based approaches.

For propagation-based approach, social media could be one major domain for news sharing, the analysis within social media relies heavily on social context features like author profiles, retweets, likes, etc. Social media rumor detection (Derczynski *et al.*, 2017) utilized conversation on Twitter to determine the veracity as RumorEval tasks. By modeling the sequence posts and behaviors on social media, researchers (Kochkina, Liakata, & Zubiaga, 2018; Ruchansky, Seo, & Liu, 2017; Volkova, Shaffer, Jang, & Hodas, 2017) proposed supervised method to detect the rumors and fake content. These approaches assume that the footprint and network of fake news are different from real news. Moreover, it has been shown that the spread speed of fake news is faster than real news (Vosoughi, Roy, & Aral, 2018). The propagation-based methods rely on social context feature; therefore, it is difficult to capture enough information for fake news detection right after the newly emerged news. Also, they are limited to social network for social context features. In contrast, this work studied reliability only on textual information, therefore, it can recognition the unreliable information in real time.

Knowledge-based method includes the tradition manual fact-checked by expert and automatic factchecking (Shi & Weninger, 2016; Shiralkar, Flammini, Menczer, & Ciampaglia, 2017; Wu, Agarwal, Li, Yang, & Yu, 2014). Several organizations, such as PolitiFact and Snopes, investigate the news and related document to report the credibility of the claim. The manual fact-checking method is time-consuming and expert oriented, which is difficult to handle the huge amount of false claim in online news media. Thus, the automated knowledge-based fact-checking system has been developed. The system will extract the claims in news content and try to match the claim to relevant data on the external knowledge base. In our work, we do not count on the external knowledge bases or web evidences; instead, we extract the stylistic features from articles to automatically capture the implicit style of unreliable article information.

Content-based methods aim to capture the keywords or writing style of malicious fabrication news from its content. The advantage of content-based methods is that it can immediately alarm the reader only from its content no matter the news is newly emerged or not. Previous works on content-based methods can be categorized into two groups by their method. One focused on the "textual content classification" (Al-Anzi & AbuZeina, 2017; Pavlinek & Podgorelec, 2017; Qu *et al.*, 2018; Wang, Luo, Li, & Wang, 2017). It classified content by "Content words", which were meaningful and different depended on the content. The other interested in "writing style recognition" (Gomez Adorno, Rios, Posadas Durán, Sidorov, & Sierra, 2018; Rexha, Kröll, Ziak, & Kern, 2018; Stamatatos, 2009) which aimed to find out the articles that have the same style but different content. These word-based methods concerned more about the "Function words" and the structure of sentence, which were often regarded as less important part before. Several research Karimi and Tang (2019); Khan, Khondaker, Iqbal, and Afroz (2019); Wang *et al.* (2018) has shown the promising result by taking advantage of machine learning technique. However, Janicka, Pszona, and Wawer (2019) address the issue that the failure of cross-domain detection, which can be interpreted as a type of overfilling on the training domain. The work conducts the experiment on four types of domain including short-text claim, full-text content. generated fake new via Amazon Mechanical Turk (AMT), and fake news on Facebook. The experiment shows that the model can fit well in the same domain, but the accuracy drops sharply when testing on the other domain.

2.2 Text Representation

To represent unique characteristics of different text documents, several features extraction methods have been proposed. Before the widespread use of the deep learning models, there are many methods relied on the hand-crafted, lexicon-based and syntactic approaches.

The hand-craft approaches are based on predefined dictionaries or linguistic resources such as the linguistic inquiry and word count (LIWC) affect lexicon (Pennebaker, Booth, & Francis, 2007). One of the advantages of using predefined dictionaries is that they are usually of high quality due to the rigorous process of labeling. However, this also presents a scalability problem as these features may not be representative of the dynamically evolving language used.

The lexicon-based approaches automatically extract the representative tokens from corpus, such as bag of word (BOW) or term frequency-inverse document frequency (TF-IDF). BOW learns the distribution of word usages to present the corpus. By integrating the n-grams consideration, the token units of BOW could be extended to n words as phrases rather than a single word to extract more high-level features. TF-IDF further introduces the statistical concept to reduce the importance of common tokens, such as "the" and "or". One of the benefits of the lexicon-based approach is that are robust to misspellings and the out of vocabulary (OOV) problems. However, it result in a extreme large size of vocabularies in memories and the curse of the dimensionality from the sparsity of vocabularies.

The syntactic approaches including part of speech (POS) parsing tree and graph-based word pattern, which considering the relation among the words. The POS parsing tree converts words by the POS tags and models the syntactic structure of sentence. The syntactic POS tree benefits the understanding for sentence, however, the POS tagging process relies on predefined dictionaries and may encountered OOV and not perform stably for specific terminologies or among different languages. The graph-based word pattern approaches (Argueta, Saravia, & Chen, 2015; Saravia, Liu, Huang, Wu, & Chen, 2018) analyze the hidden word relation by learning a word relation graph dynamically from the corpus. By adopting the graph analysis techniques, words that is important in the connection of graph structure could be extracted and used to construct the n-grams word patterns. As the word graph could present a longer connection of words than n-gram approaches, the hidden relations among words could be better preserved. The word pattern derived from graph structure learns the syntactic features of the corpus rather than n-grams key tokens; the syntactic word pattern is thus considered as a representation of the writing style. Although the method could learn the syntactic writing styles from word relation graph, however, the current approaches only focused on the English corpus. This work aims to leverage the benefits of word relation graph and propose the modification to extract syntactic writing style features from Mandarin corpus.

In the deep learning approaches, words are embedded as the vector representations by different contextual learning techniques, such as word2vec (Mikolov, Chen, Corrado, & Dean, 2013) and GloVE (Pennington, Socher, & Manning, 2014). The word vectors preserve the semantic reasoning capabilities of the word and are treated as the input feature representations to the deep learning models, such as the sequence-modeling recurrent neural network (RNN) and the convolution neural network (CNN) which focus on the local pattern extraction.

By integrating the traditional methods and the modern neural network approaches, this study proposes an approach that leverages the graph pattern features and a convolutional neural network model to identify the unreliable text information. The proposed model not only captures the textual and stylistic feature from articles but also has the adaptability for different writing styles.

3. Contextualized Affect Representation for Implicit Style Recognition

To prevent keyword bias, we studied various writing styles with a focus on frequent word usages and corresponding co-located words for each writing style. In this work, we adapted the concept of graph-based pattern extraction approaches to dynamically learn the writing style of Mandarin product review datasets. This approach has been applied in related works on emotion analysis by extracting the word patterns for each emotion. In the following sections, we highlight the adaptation of the graph-based emotion pattern approach to extract stylistic word patterns as the writing style.

The overall framework, which can be separated into stylistic pattern feature extraction (titles highlighted in orange) and model architecture (title highlighted in yellow), is shown in Figure 1. By constructing the word relation graph, the hidden word relations are preserved to enrich the stylistic words patterns in comparison to traditional lexicon-based approaches. A weighting mechanism was proposed to learn the significance of each pattern for each style.

Articles were first transformed into stylistic patterns by encoding each matched pattern and determining the corresponding score vector, which represents the article's stylistic pattern. In this work, the pattern representations were treated as the input of a neural network model for document classification based on writing style features. The details of the stylistic pattern feature extraction and model architecture are summarized in the following subsections.

4. Stylistic Pattern Features Extraction

4.1 Stylistic Graph Construction

Given a set of corpuses $\boldsymbol{C} = \{c\}$ and the sentences $\boldsymbol{S_c}$ in corpus c, the sequences of word are denoted as V_{s_c} in sentence s_c. The word graph G_c then represents the graph structure for the corpus set $\boldsymbol{C}$, such that G_c= (V_{C}, E_{C}, W_{C}). Vertices $\boldsymbol{V_C}$ is a set of nodes which represent all the word tokens v in corpus $\boldsymbol{C}$, and $A_{\boldsymbol{C}}$ is a set of arcs that represents a bi-gram relationship between each two adjacent tokens. For example, the tokenized sentence "用 _ 起來 _ 還有 _ 飾色 _ 效果 _ ， _ 給 _ 你 _ 無可取代 _ 的 _ 透亮 _ 蘋果光 _ 唷 _ ！ ！" could construct the following bi-gram relations: "用 → 起來", "起來 → 還有", "還有 → 飾色", ..., "蘋果光 → 唷", "唷 → ！ ！". Note that the under-dash "_" shows how the sentence is tokenized and the arrow "→" denotes the link relation in the word graph.

For the edge weights W, instead of initialized with binary representation, which is align with the adjacency matrix, the edge weight w_{v_i,v_j} are defined as the bi-gram probability between two word tokens v_{i} and v_{j} in order to capture the significance of link relation. The bi-gram probability is designed with a denominator of global bi-gram frequency, the frequency of all the bi-grams, rather than the degree of word node v_i or the frequency of out nodes v_{j} from node v_i. By comparing to all the bi-gram tokens, the word graph could better capture and compare the global significance for each node. Consistent to the setting of edge weight, the weighted adjacency matrix $\boldsymbol{M}$ is designed as the matrix representation of the edge weights $\boldsymbol{W}$ and defined in Definition 1.

By having the weighted mechanism, the word graph G_c could have a better ability to preserve the syntactic structure of words by a graph representation.

Definition 1 ***(Weighted Adjacency Matrix)*** *Let* $\boldsymbol{M}$ *be the weighted adjacency matrix that each entry* $M_{i,j}$ *represents the relation of word pair in the word graph* G:

$$M_{\mathrm{i,j}} = \frac{\mathrm{freq}(\ v_i,v_j)}{\sum_{v_k,v_l \in V,k\neq l} freq(v_k,v_l)} \tag{1}$$

where the freq() *denotes the frequency of two bi-gram words* v_i, v_j *or* v_k, v_l.

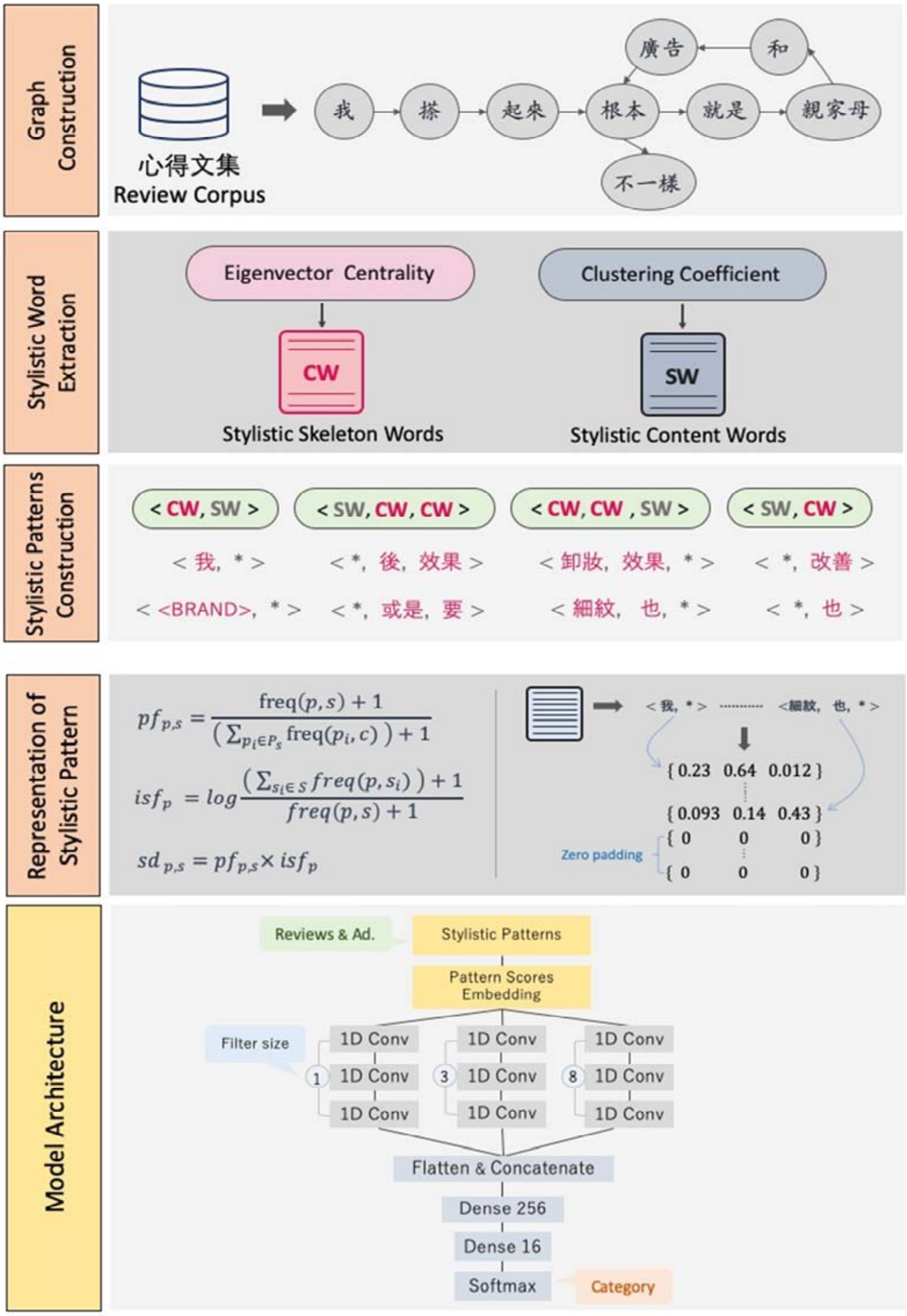

Figure 1. The framework of CARISR.

4.2 Stylistic Word Extraction

Writing styles vary from individual to individual. The idea that people utilize different distributions of words for different topics has widely been accepted in several topical methods, such as latent dirichlet allocation (LDA) (Blei, Ng, & Jordan, 2003). This work also uses this concept to extract and decompose the writing style into two elements: the stylistic skeleton and the stylistic contents. This work assumes that sentence and corpus are constructed by choosing the words of selected style to form skeleton and deciding the contents words to complete the sentence structure.

To extract the stylistic elements, two types of graph analyses—centrality and clustering—were applied to the word graph G_c. Each analysis method helps to generate a set of words: stylistic skeleton words *(CW)* (i.e., stylistic stop words) and stylistic content words *(SW)*.

4.2.1 Stylistic Skeleton

The stylistic skeleton represents the fundamental elements of word usages in a style, where such words should be widely used in all the corpuses of a given style. That is, all of the words included in the stylistic skeleton of a style should consistently appear in all of the corpuses of that style. In the structure of the graphical representation, skeleton words that represent a strong connection to other words are considered suitable candidates for stylistic skeleton words, as those words act as the fundamental nodes in the word relation graph $G_{\boldsymbol{C}}$. Inspired by Google's PageRank (Page, Brin, Motwani, & Winograd, 1999), in which nodes with high connection word nodes contribute more importance than low connection word nodes, the eigenvector centrality was selected to measure the influence of each node in $G_{\boldsymbol{C}}$.

Definition 2 ***(Eigenvector Centrality)*** *The eigenvector centrality is calculated as*:

$$e_i = \frac{1}{\lambda} \sum_{j \in \boldsymbol{V_C}} M_{i,j}\, e_j \tag{2}$$

where λ is a proportionality factor and e_i is the centrality score of word node v_i. Let λ be the corresponding eigenvalue, the equation could be rewritten into vector form Me = λe, where e is the eigenvector of M.

A word is selected as a connecter word if its eigenvector centrality e_i is higher than the empirically defined threshold θ_{eig} to ensure the quality of the high connectivity word. The higher the centrality e_i of a word v_i, the more important the word is in the graph $G_{\boldsymbol{C}}$. By the centrality measurement, a set of connector words with both high frequency and connectivity to other high-rank nodes are extracted from the word relation graph G_C and considered stylistic skeleton words $\boldsymbol{CW}$, such that $\boldsymbol{CW} = \{cw \mid e_{cw} > \theta_{eig}\}, cw \in V_C$. The examples of the stylistic skeleton words in this task (the makeup advertisement dataset) were as follows: "我," "的," "因為," "肌膚," and "特別." The extracted stylistic skeleton words not only contained numerous traditional stopwords but also style-specific words, which are known as stylistic stopwords.

4.2.2 Stylistic Content

The stylistic contents represent frequently appearing topics within a style, where topics could be formed by several separated words (i.e., LDA) or continuous word sequences. Apart from the skeleton, a topic could be presented by using the words in different ways; however, to represent the similar semantics of the topic, the topic words are generally interchangeable. For example, in the makeup advertisement dataset, there are several ways to describe the product' s effect on skin care, such as "能 _ 有效 _ 保養 _ 肌膚,""保護 _ 嫩白 _ 肌膚," or "擁有 _ 水嫩 _ 臉頰." In the above example, some word tokens can be changed while keeping the meaning the same, such as "保養" to "保護" or "嫩白" to "水嫩" and so on.

To capture the stylistic content cues, this work focuses on interchangeable word usages. By converting the style corpus in the word relation graph, the cross connections between these interchangeable word nodes are discovered. Such stylistic content word nodes tend to cluster with other nodes that share this or similar concepts. The clustering behavior in the graph can be measured by a graph analysis factor, namely the *clustering coefficient*, which determines how a node interconnects with its neighbor nodes. This work therefore applied the *clustering coefficient* to dynamically extract the stylistic contents, as shown below.

Definition 3 ***(Clustering Coefficient)*** *The clustering coefficient is defined by clustering coefficient as:*

$$cl_i = \frac{\sum_{j\neq i,k\neq j,k\neq i,} M_{i,j} \times M_{i,k} \times M_{j,k}}{\sum_{j\neq i,k\neq j,k\neq i,} M_{i,j} \times M_{i,k}} \times \frac{1}{|\boldsymbol{V_C}|} \tag{3}$$

where cl_i *denotes the average clustering coefficient of node* v_i*.*

Similarly, the word nodes v_i were also filtered by a predefined threshold θ_{tri} for clustering coefficient cl_i to ensure the clustering quality. During the computing process of clustering coefficient cl_i for each node v_i, we discovered that there were many nodes with high coefficients. However, many of them belonged to local mini-clusters in which the degree of node was too small, resulting in too many specific words for stylistic contents. A post-filtering step was then applied to remove the local mini-cluster and small cluster words based on the number of triangles tri_i of the word nodes v_i, where less node triangles indicated a smaller cluster. With the post-filtering step, a set of qualified stylistic content words SW were retrieved, such that $\boldsymbol{SW} = \{sw \mid cl_{sw} > \theta_{cl},\ tri_{sw} > \theta_{tri}\},\ sw \in \boldsymbol{V_C}$, where θ_{tri} denotes the empirical threshold for the number of triangles for the word node. Some examples of stylistic content words in this task were "森林系," "世界級," "黏稠度," and "可愛感."

4.3 Stylistic Pattern Construction

With the extracted stylistic skeleton and stylistic content words, this step aimed to construct the stylistic word pattern template. The stylistic word pattern is designed to capture hidden word usages in a writing style. For a word pattern, the length *l* of the pattern can be dynamic; that is, there may exist a longer stylistic word pattern (i.e., slogans) or a shorter one (i.e., topic tokens). In this work, a short length was adapted, as a longer word pattern may be difficult to match in a real-world case.

To construct the word pattern templates $\boldsymbol{P} = \{p\}$, the permutation of stylistic skeleton and content words, *CW* and *SW*, were adopted in our work using the rules below:

- The stylistic skeleton words are required to exist in the pattern at any position as such words have the top connectivity in the corpus.
- A word pattern could contain more than one skeleton words.

For example, in pattern length $l = 3$, each pattern feature is composed of an arbitrary permutation, such as "cw sw cw" or "cw sw sw," from the set of *CW* and *SW*. The word patterns are then used to search the corpus set $\boldsymbol{C}$ to retrieve the pattern frequency. The word patterns that belongs to last 20% infrequent patterns are dropped, as they are not general enough.

Instead of utilizing the word pattern by exact matching (bag-of-word matching) as n-gram does, this work adopts a flexible representation to increase the versatility of the pattern template due to the issue of easily overfitting for n-grams and pattern size consumption. Compared to the stylistic skeleton words, the stylistic content words are relatively easier to update or replace (i.e., develop new terms) as these are determined by the clustering coefficient, which captures interchangeable words. With respect to the stylistic content characteristics, various words that may be beyond the knowledge coverage of the training dataset could be used to describe a topic. Therefore, flexible representation was designed and performed by replacing the *SW* in the word pattern with a placeholder <*>, which means any token could be considered in the stylistic patterns during the matching process (i.e., "我 <*> 肌膚", "特別 <*> 的").

The flexibility of the pattern (the wildcard representation <*>) enables our model to possess robust generalization ability, which increases pattern coverage for dealing with out-of-vocabulary words and slang or coded words used in specific domains when extracting features during testing. The complete steps for stylistic word extraction and stylistic pattern construction are formally summarized in Algorithm 1.

Algorithm 1 Stylistic Pattern Features Extraction Algorithm

```
Calculate eigenvector centrality ($e$) and clustering coefficient ($cl$) for topic graph.
Set $\theta_{eig}$, $\theta_{cl}$, $\theta_{tri}$ thresholds of centrality, clustering coefficient and number of triangles.
CW← a set of stylistic skeleton words
TW← a set of stylistic content words
for all node $v$ in V do
  $tri_v$= number of triangles for $v$
  if $e_v$> $\theta_{eig}$ then
    CW← $v$ end if if $cl_v$> $\theta_{cl}$ and $tri_v$> $\theta_{tri}$ then
    SW← $v$
  end if
end for
Construct patterns P with the permutation of stylistic skeleton words and content words.
for all pattern $p$ in P do
  $p$ = Replace the $sw$ with wildcard (<*>) from $p$
end for
```

4.4 Representation of Stylistic Pattern

With the stylistic word pattern, it is critical that how to transform a set of patterns to features for the classification. One of the traditional ways is to present the word pattern as a set of bag-of-patterns with the frequency or normalized frequency (probability of occurrence) as the numerical features. However, such bag-of-pattern representations limited in the current state-of-the-art deep neural network (DNN) models, which applied several word embedding techniques to present the hidden information for a word. Such embedding features are very flexible which could be utilized not only in traditional classifiers (i.e. support vector machine (SVM) or random forest), but also the DNN models.

Inspired from it, this work aims to proposed a flexible numerical vector representation for the extracted word patterns in a pre-training manner which could perform as the initialized parameters for the classification models. The numerical representation is designed to leverage the uniqueness of each word pattern for each label, which is the style in this work. The uniqueness of the pattern for different labels is calculated by a weighting schema, namely *identical stylistic degree*. Formally, given a set of corpuses $\boldsymbol{C} = \{c\}$ and a set of possible style $\boldsymbol{S} = \{s\}$, where each corpus *c* belongs to a style *s*, the identical stylistic degree is defined by three components, which are *pattern frequency*, *inverse style frequency*.

Definition 4 ***(Pattern Frequency)*** *The pattern frequency pf is defined as:*

$$pf_{p,s} = log\frac{\text{freq}(p,s)+1}{1+\sum_{p_i \in \boldsymbol{P_s}} \text{freq}(p_i,s)} \tag{4}$$

where $\text{freq}(p,s)$ *represents the frequency of the pattern p in the style s, and* $pf_{p,s}$ *is the logarithmic scaled frequency of p in all the articles of the style s.*

Pattern frequency is designed to capture the frequently appeared word pattern under the assumption that the more a pattern exists in the corpus of a style, the more important the pattern is. As the frequency is dramatically different from pattern to pattern, the scale of the $\text{freq}(p,s)$ score may encounter biased due to the large frequency gap. A logarithm function is thus applied to avoid the identical stylistic degree dominated by pattern frequency.

Definition 5 ***(Inverse Style Frequency)*** *The inverse style frequency* isf *is computed as:*

$$isf_p = log\frac{1+\sum_{s_i \in \boldsymbol{S}} \text{freq}(p,s_i)}{\text{freq}(p,s)+1} \tag{5}$$

where isf_p *is the measurement of the rareness of the pattern p in all articles.*

The inverse style frequency aims to decrease the importance for the commonly appeared pattern among many styles. The traditional inverse document frequency in TF-IDF is designed to examine whether the pattern exist in how many styles. However, the pattern frequency in a style is able to be treated as the intensity of the pattern existence. This work then refines the inverse style frequency by introducing the pattern frequency as indicator to calculate the cross styles uniqueness.

Finally, the uniqueness of each stylistic pattern could be presented by the identical stylistic degree as below.

Definition 6 ***(Identical Stylistic Degree)*** *The identical stylistic degree sd is calculated as:*

$$sd_{p,s} = pf_{p,s} \times isf_p \tag{6}$$

where $sd_{p,s}$ *is the identical stylistic degree that represents the importance of the pattern* p *to the style s.*

With the identical stylistic degree $sd_{p,s}$, it is able to quantify the uniqueness of each stylistic word pattern p for a style s. The stylistic pattern p is then able to present in a vectorized form $X_p = |\ sd_{p,s}\ |$, $X_p \in R^{|\boldsymbol{S}|}$, namely stylistic pattern embeddings, where each component represents the identical stylistic degree $sd_{p,s}$ of pattern p for a style s. The flexibility of the proposed identical stylistic degree also allows the weighting schema to be extended when the number of styles $|\ \boldsymbol{S}\ |$ is increased.

5. Model Training

In this section, we describe the classification model and the transfer learning procedure.

5.1 Model Architecture

Due to the well performance of Convolutional Neural Network architecture on several text classification tasks in the past, CARISR was based on Multi-layer ConvNet (Kim, 2014) architecture, as shown in the bottom of Figure 1. Consider a set of corpuses $C = \{c_1, c_2, \dots, c_n, \dots c_N\}$, where $n \in [1, N]$. Each article c_n was transformed into pattern degree matrix X_n based on the stylistic pattern embedding described in previous section.

$$X_n = PatternEmbedding(c_n), \text{where } X_n \in R^{L\times|\boldsymbol{C}|} \tag{7}$$

where L denotes the parameter as the threshold for the maximum number of patterns for an article, and $|\boldsymbol{C}|$ denotes the number of categories, respectively. If the number pattern for an article is less than L, it will be filled with zero as pattern scores. For the sake of brevity, we used X to present single instance X_n. Each entry $X_{i,j}$ in the pattern degree matrix X represented identical stylistic degree for pattern i in category j, where $i \in [1, |\boldsymbol{C}|], j \in [1, L]$.

X is following fed into three paths which are composed by 1-D convolutional layer with different filter size of 1, 3, and 8. The output is passed through a ReLU activation function (Nair & Hinton, 2010) that produces a feature map. A 1-D max pooling layer of size 3 is then applied to each feature map.

$$a_i = ReLU(conv(X, filter\_size = i)) \tag{8}$$

$$\widehat{a_\iota} = MaxPooling(a_i) \tag{9}$$

the above two steps are simplified as following equation:

$$\widehat{a_i} = conv\_block(X, i) \tag{10}$$

where i denotes filter size. Stacked with three $conv\_block$, the results were concatenated together and passed through two fully connected layers of dimensions 256 and 16 in order.

$$a = \widehat{a_1} \oplus \widehat{a_3} \oplus \widehat{a_8} \tag{11}$$

$$d_1 = ReLU(W_a a + b_a) \tag{12}$$

$$\text{Classification: } s = softmax(W_d d_1 + b_d) \tag{13}$$

where $\oplus$ denotes the concatenate operation, $\widehat{a_i}$ is the output of stacked block which kernel size is i. We used softmax to get the probability of each category and used cross entropy as loss function. In order to prevent overfitting to training data, Dropout was applied to convolution layers and fully connected layers. The corresponding dropout rate is 0.5 and 0.7. The L2 regularization is also applied in the loss function, and the coefficient is 0.05. We chose a batch size of 64 and trained for 12 epochs using Adam optimizer (Kingma & Ba, 2014). We used Keras (Chollet *et al.*, 2015) to implement the CARISR architecture.

5.2 Transfer Learning

Due to the difficulty of collecting labelled sponsored reviews and self-purchased product reviews, a limited dataset was available to train the classifier to distinguish sponsored reviews from self-purchased product reviews. Inspired by the idea of transfer learning, we predicted that the flexibility of the proposed stylistic patterns could enable the proposed model to be transferable. This research thus proposes a two-stage training process to recognize sponsored reviews.

In the first stage, a large amount of advertisement and product review data were collected as weak label data to pre-train the CARISR model. In terms of writing styles, advertisements are designed to highlight the features of sale products, while sponsored reviews are written in a manner similar to trial reviews. However, sponsored reviews are considered a special kind of advertisement, as they aim to both introduce the product and spotlight it. More specifically, both advertisements and sponsored reviews have the same objective, which is to advertise the product in a positive manner. In other words, the model could learn the diverse writing styles of advertisements in the early stages (learning from advertisement) through the weak label pre-trained procedure.

In the second stage, the transfer learning concept was applied to fine-tune the pre-trained model with what little sponsored review data were available. Having the prior knowledge of the advertisement writing style, the model could more easily learn to distinguish sponsored reviews. To fine-tune it, the parameters of CNN blocks were fixed, and the first fully connected layer in CARISR was taken as the feature vector of articles. The feature vector was fed into another fully connected layer to examine the transformation from feature vector to classification result. This approach allows CARISR to distinguish sponsored reviews from true product reviews.

In this two-stage transfer learning process, the model's feature representation improved thanks to pre-training with a large amount of weak label data. It learned to distinguish the writing style of sponsored reviews and product reviews through fine-tuning with the small amount of true label data available. Based on the training process, we predict that even with the lack of true labeled data, the model could still perform well and avoid overfitting.

6. Experiments

6.1 Data

To distinguish the sponsored and product review, this research utilized the transfer learning concept which leveraged user reviews and advertisement articles as pre-training corpus and fine-tune the model with sponsored and self-purchased product reviews. For the entire training process, two datasets are collected and introduced below.

The first dataset was collected from UrCosme, a famous makeup product review website in Taiwan, with three classes *Self-purchased product review*, *Trial product review*, and *Advertisement*, where the three classes are tagged and verified by UrCosme. It has total 194,099 makeup reviews from 17,006 users from 2015 to 2018 June and includes 22,094 products and 4,594 articles from 498 brands.

The second dataset was from PIXNET, an online social blog in Taiwan, makeup product-related articles are collected with three classes Self-purchased product review, Trial product review, and the target Sponsored review. Since there are no article tags provided from PIXNET, several rules are defined for identifying the three classes. Firstly, the Sponsored review are the articles which contain the URL links with specific blogger's identification tokens. To trace the web reference from which bloggers to the product web page, this kind of URLs are widely been used to record the number of clicks and make profits to the bloggers. The text content from articles with specific URLs are collected with the Sponsored review label. Second, based on matching the keywords, "邀稿" and "試用", to label the Trial product review and other normal product reviews are labeled as Self-purchased product review. After categorizing the articles, we manually pick 125 articles from each category as the PIXNET dataset and cross valid the dataset with 5 experts. To prevent our model learned from the specific contents, all the clues (including URLs and keywords, tokens that have used to create labels) are removed in advance.

Due to the lack of the sponsored review, the UrCosme dataset is considered as the weak label dataset for the main task, the classification of sponsored and product review. The PIXNET dataset is treated as the ground truth dataset as it is labeled by manual efforts. The detail data distribution of two datasets are shown in Table 1 and Table 2. The experiment 6.3 takes the training part of the UrCosme dataset for model pre-training but evaluates on the testing part of PIXNET dataset. In experiment 6.4, the completed PIXNET dataset is involved for evaluating the pre-training model from UrCosme dataset. For experiment 6.5, the PIXNET dataset is down sampled following the ratio 4:1 for fine-tuning and evaluating.

Table 1. The data distribution of UrCosme dataset.

	Total	Training	Testing
Advertisement	9,681	9,681	2,423
Trial product review	87,508	10,000	2,423
Self-purchased product review	106,591	10,000	2,423

Table 2. The data distribution of manual labeled PIXNET dataset.

	Total
Sponsored review	125
Trial product review	125
Self-purchased product review	125

6.2 Baseline Methods

To represent a text corpus, the term frequency-inverse document frequency (TF-IDF) has been widely used in several text classification tasks. It could automatically learn the important n-grams from the corpus and present the corpus based on the extracted important n-grams. Represented by the TF-IDF features, all the articles were transformed into TF-IDF feature vector with 2500 dimensions for the extraction of the important n-grams.

In deep neural network (DNN) approaches, a text corpus is frequently represented by a sequence of the word vectors, namely *word embeddings*. The word embeddings could be either provided by a pre-trained word vectors or derived by the DNN models during the training procedure. In this work, a pretrained 400 dimensions word vector from *YZU NLP Lab*[1], trained from traditional Mandarin Wikipedia, were applied as initialized representation to present the words. The word embeddings were set as trainable to be fine-tuned in the learning procedure.

For the classification model, both traditional model and DNN model were applied in our

[1] http://nlp.innobic.yzu.edu.tw/demo/word-embedding.html

work, which were the Logistic Regression (LR) model and the Long Short-term Memory (LSTM) model. The LR model learned a specific weight for each dimension of the features, which could provide a more interpretable explanation for analysis. For DNN models, the text-CNN and LSTM were applied in the experiments. The text-CNN (Kim, 2014) considers local word features by *n*-gram windows. By adopting multiple convolutional layer, model could summarize the local word features and representation the corpus. This work set the filter size of convolution layer as 3, stacked 3 convolution layers and following with 512,128 dense layers for feature summary. The LSTM model takes the input word sequence in a word by word manner and models the words relation step by step. In this work, the bi-directional LSTM with attention mechanism was applied which achieved several state-of-the-art performance for many NLP tasks. The LSTM model was connected with a 128-dimension fully connected layer for feature summary. For two DNN models, the categorical predictions were done by the Softmax activation function for feature summaries.

6.3 Weak Label Classification Training

In the first training stage, all of the models were trained to distinguish the three different classes with the UrCosme dataset as weak label pre-training for the main task, which was the classification of sponsored and product reviews. After the model pre-training, the testing data from UrCosme was applied to evaluate the pre-training performance, the results of which are shown in Table 3. Overall, the proposed CARISR did not have the best performance in the first stage of the training process compared to the TF-IDF baseline method and LSTM-based models. However, after analyzing the weight of the model, we observed that the baseline method result was easily influenced by specific keywords. An example from a real article is discussed below:

感謝 UrCosme 與 SK-II，讓我參與「超肌因鑽光淨白精華」新品活動！

超肌因鑽光淨白精華 0.7ml x 28 包使用方式

・於清潔肌膚後，先使用 SK-II 青春露調理肌質，有效提升細滑度、緊緻度、抗皺度、白皙度、光澤度等五大美肌度。

・ 接著 ...... 乳白色精華無特別香氣，它使肌膚好吸收無黏膩，說實在的，當每晚保養擦上精華後，我都覺得肌膚看起來變得平滑、有光澤、膚質超好的，總覺得它有美肌般的效果！連續使用幾天，肌膚的黯沉、泛黃有改善，轉為明亮、光澤度大大提升，真心滿意，會想買正貨！

Thanks for UrCosme and SK-II for inviting me to join this campaign!

How to use SK-II Facial Treatment Essence 0.7ml * 28

> *After cleaning the face, apply SK-II Facial Treatment Essence can keep your face moisturized, brighten and firming.*
>
> *then… ...it makes my skin without stinging, literally, once applied the essence, it spreads easily and gets absorb quickly into the skin, besides, my skin felt moisturized without any greasy feeling. Continuing using for 2 weeks, my skin feels more brighten and firmer. I am really satisfied with this product and will order again once I run out!*

The example articled was a trial product review, which it was correctly classified as by the baseline models but was incorrectly classified as an advertisement by the CARISR model. Although this article was misclassified as an advertisement, the writing style of the article showed more similarity to an advertisement than a real review by human judgement. By analyzing the weight of each term in the LR model, the result showed that the model relied on some specific terms, such as activity (活動), satisfy (滿意), and invite (邀請). In this example, the model would be easily misled by malicious writers due to these specific terms.

Based on this example, although the accuracy of the CARISR model result was lower, it gave greater consideration to the relation between word structures in the article as a whole. The following experiment shows that the CARISR model was better able to resist the influence of specific terms.

Table 3. The classification result of four methods on UrCosme dataset.

Method	**Avg.F1-Score**	**Ad.F1-Score**
TF-IDF	0.79	0.97
text-CNN	0.79	0.98
bi-LSTM-attention	0.82	0.98
CARISR	0.70	0.97

6.4 Sponsored Review Testing

The pre-trained models were evaluated with the testing data from the weak labeled UrCosme dataset discussed in the previous section. The pre-trained models were evaluated with the human-labeled dataset; that is, the reviews from PIXNET were used as testing data with the advertisement label in UrCosme replaced by sponsored review label. As shown in Figure 2, although the baseline models had better performance using the pre-trained settings, they performed worse than CARISR using the PIXNET dataset. More importantly, in the classification of sponsored reviews, baseline methods could not successfully differentiate sponsored reviews. This indicates that the baseline models had a good ability to learn but were hampered by the overfitting issue when using the training dataset. The main reason for this was that the baseline methods relied heavily on specific terms as clues, which resulted in the models not being general enough to apply to different testing data, even data from the same domain dataset (in this task, both were sponsored makeup reviews). Instead, CARISR leveraged the stylistic patterns to keep the features of sentence structure and writing style rather than only specific keywords or n-grams. Therefore, even if the testing dataset was slightly changed, the model was still able to determine the advertisement writing style.

In real-world sponsored reviews, malicious writers usually pretend that the advertisement is a self-purchased product review. Many words used in commercial reviews usually appear in self-purchased product reviews; therefore, it is easy for them to avoid detection if the model relies heavily on specific terms or baseline methods. The proposed model, CARISR, was better able to avoid this problem, making it more suitable to real-world situations.

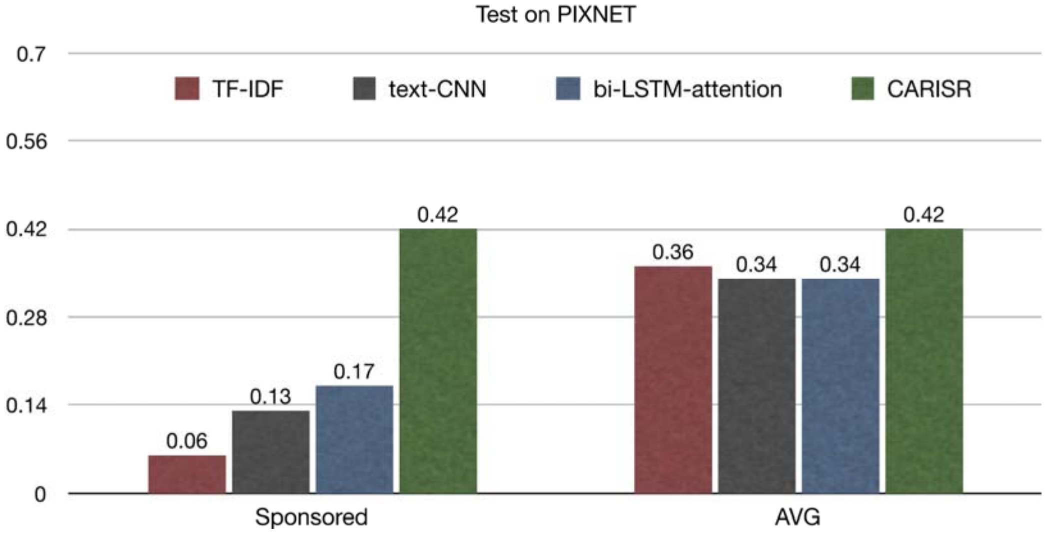

Figure 2. Comparison of TF-IDF, text-CNN, bi-LSTM-attention, and CARISR when applied to the PIXNET dataset. AVG is the average F1-score for all three categories, and Sponsored is the F1-score for sponsored reviews.

6.5 Transfer Learning with Sponsored Reviews

According to the classification results presented in the previous section, CARISR demonstrated the ability to recognize the latent writing styles of sponsored articles. Transfer learning was applied to fine-tune the DNN models to boost its performance based on a small number of manually collected sponsored reviews on PIXNET. One-fifth of the PIXNET dataset (25 samples for each class) was kept for the final testing, and the rest of the data were utilized for fine-tuning (100 samples for each class). Note that the TF-IDF model was excluded from this section, as it is not able to perform standard transfer learning based on the

TF-IDF and LR algorithms. The experimental result, labelled Transfer-3, is shown in Figure 3.

All three of the tested models manifested better performance after adjusting the parameters using transfer learning. For three-label classification, the text-CNN, bi-LSTM-attention and CARISR had F1-scores of 0.21, 0.47 and 0.51, respectively. Furthermore, our analysis found that a large percentage of collected sponsored reviews were very similar to advertisements. This may be the reason why the CARISR-Trans3 did not perform as well as expected.

Therefore, we conducted another experiment that only used sponsored reviews and self-purchased product reviews, as checked by humans, to build a binary classification model. As shown in Figure 3, with the application of two-category transfer learning (Transfer-2), the CARISR F1-score was improved to 0.70 and outperformed the bi-LSTM-attention by 0.07 points.

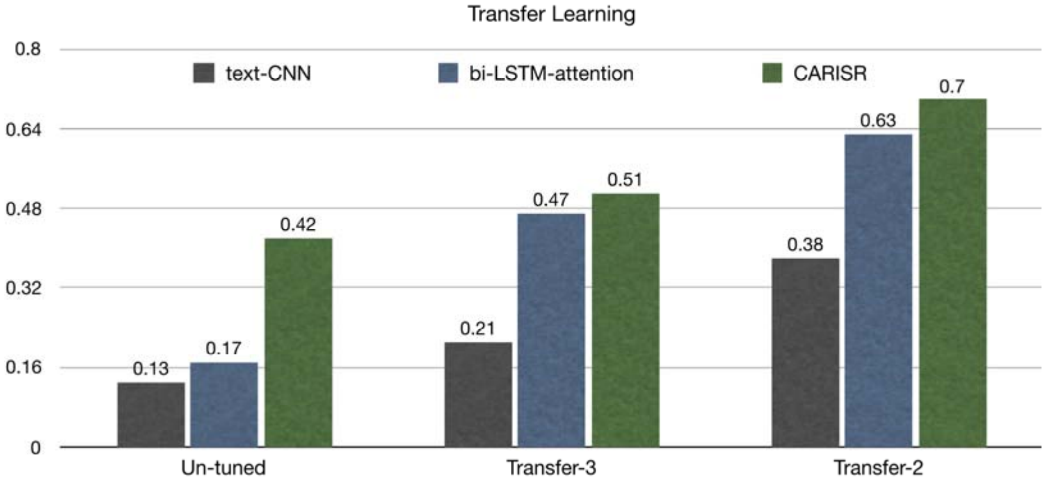

Figure 3. Comparison between original method and transfer learning. Transfer-3 indicates the result of the models after fine-tuning using three categories: sponsored, trial product, and self-purchased product review. Transfer-2 shows the results of the models after fine-tuning with only sponsored and self-purchased product reviews.

7. Conclusion

This research mainly focused on quantifying the reliability problem that results from sponsored articles on popular Mandarin forums or websites. To address the problem with limited labeled data, we first proposed a framework, CARISR, that combines weak label and transfer learning methods. CARISR can learned implicit writing styles from weak label data, and it can be further improved by transfer learning with minimal amounts of manually labelled data. Thanks to its graph-based feature, CARISR is not only more robust, but it also has better generalization compared to the traditional token-based features. Experimental results showed that our model can correctly recognize around 70% of sponsored articles from the human-labeled dataset.

Our work provides a new perspective on and further improvement to reliability tasks. In the future, we plan to merge graph-based and semantic features to capture more underlying meaning in context. Meanwhile, the enrichment of stylistic word patterns could also improve model comprehension.

References

Al-Anzi, F. S., & AbuZeina, D. (2017). Toward an enhanced arabic text classification using cosine similarity and latent semantic indexing. *Journal of King Saud University-Computer and Information Sciences*, *29*(2), 189-195. doi: 10.1016/j.jksuci.2016.04.001

Argueta, C., Saravia, E., & Chen, Y.-S. (2015). Unsupervised graph-based patterns extraction for emotion classification. In *Proceedings of the 2015 ieee/acm international conference on advances in social networks analysis and mining 2015*, 336-341. doi: 10.1145/2808797.2809419

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, *3*(Jan), 993-1022.

Chollet, F. et al. (2015). *Keras*. https://keras.io.

Derczynski, L., Bontcheva, K., Liakata, M., Procter, R., Hoi, G. W. S., & Zubiaga, A. (2017). Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours. In arXiv preprint arXiv:1704.05972.

Gomez Adorno, H. M., Rios, G., Posadas Durán, J. P., Sidorov, G., & Sierra, G. (2018). Stylometrybased approach for detecting writing style changes in literary texts. *Computación y Sistemas*, *22*(1), 47-53. doi: 10.13053/CyS-22-1-2882

Janicka, M., Pszona, M., & Wawer, A. (2019). Cross-domain failures of fake news detection. *Computación y Sistemas*, *23*(3), 1089-1097. doi: 10.13053/CyS-23-3-3281

Karimi, H., & Tang, J. (2019). Learning hierarchical discourse-level structure for fake news detection. In arXiv preprint arXiv:1903.07389.

Khan, J. Y., Khondaker, M. T. I., Iqbal, A., & Afroz, S. (2019). A benchmark study on machine learning methods for fake news detection. In arXiv preprint arXiv:1905.04749.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In arXiv preprint arXiv:1408.5882.

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

Kochkina, E., Liakata, M., & Zubiaga, A. (2018). All-in-one: Multi-task learning for rumour verification. In arXiv preprint arXiv:1806.03713.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. In arXiv preprint arXiv:1301.3781.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (icml-10)*, 807-814.

Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). *The pagerank citation ranking: Bringing order to the web*. (Technical Report No. 1999-66). Stanford InfoLab. Previous number = SIDLWP-1999-0120. Stanford InfoLab. Retrieved from http://ilpubs.stanford.edu:8090/422/

Pavlinek, M., & Podgorelec, V. (2017). Text classification method based on self-training and lda topic models. *Expert Systems with Applications*, *80*, 83-93. doi: 10.1016/j.eswa.2017.03.020

Pennebaker, J., Booth, R., & Francis, M. (2007). Linguistic inquiry and word count: Liwc [computer software]. Austin, TX: liwc. net.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)*, 1532-1543. doi: 10.3115/v1/D14-1162

Qu, Z., Song, X., Zheng, S., Wang, X., Song, X., & Li, Z. (2018). Improved bayes method based on TF-IDF feature and grade factor feature for chinese information classification. In *Proceedings of 2018 ieee international conference on Big data and smart computing (bigcomp)*, 677-680. doi: 10.1109/BigComp.2018.00124

Rexha, A., Kröll, M., Ziak, H., & Kern, R. (2018). Authorship identification of documents with high content similarity. *Scientometrics*, *115*(1), 223–237. doi: 10.1007/s11192-018-2661-6

Ruchansky, N., Seo, S., & Liu, Y. (2017). Csi: A hybrid deep model for fake news detection. In *Proceedings of the 2017 acm on conference on information and knowledge management*, 797-806. doi: 10.1145/3132847.3132877

Saravia, E., Liu, H.-C. T., Huang, Y.-H., Wu, J., & Chen, Y.-S. (2018). Carer: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, 3687-3697. doi: 10.18653/v1/d18-1404

Shi, B., & Weninger, T. (2016). Fact checking in heterogeneous information networks. In *Proceedings of the 25th international conference companion on world wide web*, 101-102. doi: 10.1145/2872518.2889354

Shiralkar, P., Flammini, A., Menczer, F., & Ciampaglia, G. L. (2017). Finding streams in knowledge graphs to support fact checking. In *Proceedings of 2017 ieee international conference on data mining (icdm)*, 859-864. doi: 10.1109/ICDM.2017.105

Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, *60*(3), 538-556. doi: 10.1002/asi.21001

Volkova, S., Shaffer, K., Jang, J. Y., & Hodas, N. (2017). Separating facts from fiction: Linguistic models to classify suspicious and trusted news posts on twitter. In *Proceedings of the 55th annual meeting of the association for computational linguistics*,volume 2: Short papers, 647-653. doi: 10.18653/v1/P17-2102

Vosoughi, S., Roy, D., & Aral, S. (2018). The spread of true and false news online. *Science*, *359*(6380), 1146–1151. doi : 10.1126/science.aap9559

Wang, T., Luo, T., Li, J., & Wang, C. (2017). Reasearch on feature mapping based on labels information in multi-label text classification. In *Proceedings of 2017 7th ieee international conference on Electronics information and emergency communication (iceiec)*, 452-456. doi: 10.1109/ICEIEC.2017.8076603

Wang, Y., Ma, F., Jin, Z., Yuan, Y., Xun, G., Jha, K., … Gao, J. (2018). Eann: Event adversarial neural networks for multi-modal fake news detection. In *Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining*, 849-857. doi :10.1145/3219819.3219903

Wu, Y., Agarwal, P. K., Li, C., Yang, J., & Yu, C. (2014). Toward computational fact-checking. *Proceedings of the VLDB Endowment*, *7*(7), 589-600. doi: 10.14778/2732286.2732295

UrCosme. Retrieved July 18, 2018, from https://www.urcosme.com/

PIXNET. Retrieved July 18, 2018, from https://www.pixnet.net/