Information Processing and Management 58 (2021) 102474

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

DOFM: Domain Feature Miner for robust extractive summarization

Hiren Kumar Thakkar [a,b], Prasan Kumar Sahoo [a,c,*], Pranab Mohanty [d]

[a] *Department of Computer Science and Information Engineering, Chang Gung University, Guishan, 33302, Taiwan*
[b] *Department of Computer Science Engineering, School of Engineering and Applied Sciences, Bennett University, Greater Noida, 201310, India*
[c] *Division of Colon and Rectal Surgery, Chang Gung Memorial Hospital, Linkou, 33305, Taiwan*
[d] *Center of Excellence in AI, Fidelity Investments, Boston, MA, USA*

ARTICLE INFO

Keywords:
Opinion feature mining
Domain
Sentiment analysis
Extractive summarization

ABSTRACT

The domain feature retrieval has potential applications in text summarization. However, it is challenging to mine domain features from the user reviews. In this paper, a novel Domain Feature Miner (DOFM) is designed by (i) formulating the feature mining problem as a clustering problem and (ii) engaging three newly conceived empirical observations such as frequency count, grouping semantics, and distributional statistics of features. Later, Symmetric Cluster Extraction (SCE) and Asymmetric Cluster Extraction (ACE) algorithms are designed to identify domain features from clusters. The effectiveness of the DOFM is verified on benchmarks provided by the University of Illinois at Urbana–Champaign and compared with the four state-of-the-art (SOTA) approaches using Precision, Recall, and F-score. Moreover, ROUGE (Recall-Oriented Understudy for Gisting Evaluation), a well-known package for automatic evaluation of summaries is used to evaluate the DOFM generated summaries. The Error Analysis reveals that at least one of three annotators would prefer 84% sentences of all DOFM generated summaries, while 36% sentences are preferred by all three. This indicates the robustness of DOFM in domain feature retrieval and extractive summarization.

1. Introduction

The domain features retrieval from user reviews has potential applications in summarization (Amplayo & Song, 2017; Nasar, Jaffry, & Malik, 2019), sentiment analysis (Xu, Pan, & Xia, 2020), and recommendation system (García-Sánchez, Colomo-Palacios, & Valencia-García, 2020) etc. The Extractive Summarization (ES) is one of such applications, where relevant sentences are selected to generate meaningful summary. The existing ES methods select the sentences using key phrases, sentence length, sentence position etc. However, the aforementioned parameters do not guarantee the sentences with domain features. Such summaries are least helpful to the stakeholders. Therefore, there is a need for a robust system that identifies domain features from colloquial user reviews. Normally, user reviews are comprised of domain and sentiment features as given in Example 1.

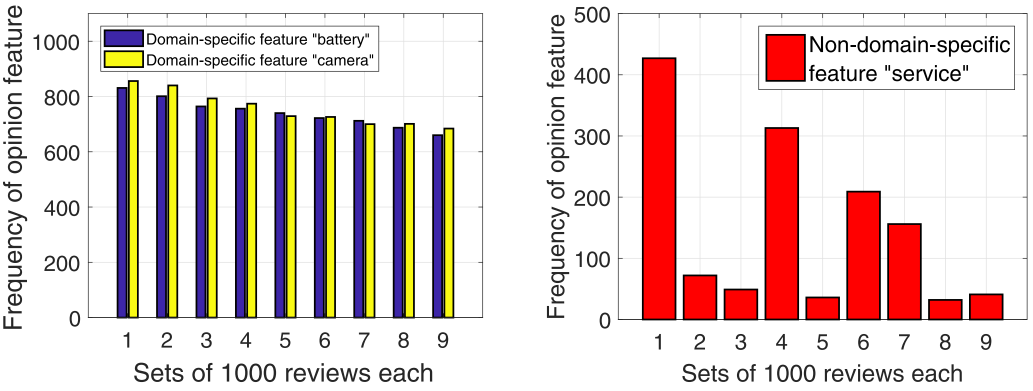
Example 1. The cellphone *battery* is the *best*.

Here, the "battery" is a domain feature and "best" is a sentiment feature. Sentiment feature extraction problem is relatively easy with sentiment dictionaries (Wu, Wu, Chang, Wu, & Huang, 2019). However, domain feature dictionaries are not readily available and need to be mined from the user reviews.

Usually, the Naïve approach is to designate frequent terms are domain features (Hu & Liu, 2004). For example, terms "battery" and "camera" appear frequently in "cellphone" reviews and therefore designated as domain features. However, Naïve approaches

[*] Corresponding author at: Department of Computer Science and Information Engineering, Chang Gung University, Guishan, 33302, Taiwan.
E-mail addresses: hirenkumar.thakkar@bennett.edu.in (H.K. Thakkar), pksahoo@mail.cgu.edu.tw (P.K. Sahoo), pranabmohanty@gmail.com (P. Mohanty).

https://doi.org/10.1016/j.ipm.2020.102474
Received 11 July 2020; Received in revised form 4 December 2020; Accepted 15 December 2020



(a) (b)

Fig. 1. Frequency distribution of (a). domain opinion feature "battery", and (b). Non-domain opinion feature "service", in cellphone data set (Wang, Lu, & Zhai, 2011).

have two shortcomings. First, the selection of frequent but inaccurate domain features. Second, the rejection of valid but overly general domain features. For instance, term "amazon" appears frequently in user reviews from Amazon portal, which is an invalid domain feature. On the contrary, term "price" is a valid "cellphone" feature. However, it is discarded as it is overly general term (Hai, Chang, Kim, & Yang, 2014).

Any domain feature retrieval system faces the tradeoff between the accurate feature retrieval and system complexity. The Naïve approaches are relatively easy to implement, but they often capture a large number of inaccurate features as well. On the contrary, deep learning and supervised learning-based approaches perform well, but they incur additional computational costs and require the training of data. Therefore, a low-cost unsupervised Domain Feature Miner (DOFM) is designed to balance the tradeoff. The DOFM reduces the system complexity by incorporating three empirical observations. (1) The domain features appear in several reviews. (2) The domain features appear together and form a strong grouping-semantics. (3). The frequency distribution of domain features is consistent across the reviews.

The motivation behind unsupervised clustering based on the frequent co-occurring candidate features is evident from Fig. 1a and b. As evident from Fig. 1a, "battery" and "camera" are cellphone-specific features and co-appear in several user reviews on an experiment on 9000 user reviews. On the contrary, non-cellphone-specific feature "service" has a scattered presence, which hardly co-appear with other features. Based on our key findings, the following are the research objectives.

1.1. Research objectives

- To identify the general but valid features and ignore the frequent but invalid features.
- To identify the valid domain features by designing an unsupervised yet simple approach to reduce the system complexity.
- To validate the effectiveness of key empirical observations such as frequency count, grouping semantics, and distribution statistics for mining rich opinion features.
- To validate the impact of data set size and review size on the performance of DOFM system.
- To verify the goodness of extractive summaries generated using DOFM retrieved features.
- To perform the rigorous quality assessment of DOFM generated features to the baseline and state-of-the-art (SOTA) existing approaches.

The DOFM brings the novelty by engaging three simple, scalable yet less compute-intensive empirical observations such as frequency count, grouping semantics, and distributional statistics of features. Employing the empirical observations to formulate the unsupervised clusters form the basis of high quality features ignoring frequent but inaccurate features and thereby retain the rich feature set. Moreover, the unsupervised clustering approach greatly simplifies the feature extraction process as it eliminates the need for expensive manual feature labeling task and also eliminates the training of the model on a huge data set. Contrary to a few existing studies that extract domain features using syntactic and semantic rules, which are difficult to generalize and limit the applicability, the DOFM iteratively formulates the symmetric and asymmetric unsupervised clusters of features. The grouping semantics formulates the clusters that help DOFM to effectively discard the frequent yet invalid features. Another novelty of DOFM is it is supported by symmetric and asymmetric cluster extraction algorithms, which are proved scalable and less compute-intensive using theorems in the proposed work.

The DOFM system is summarized as follow. First, a set of candidate features are obtained from user reviews by discarding the stop words and clustered based on the key findings. Later, the customized Symmetric Cluster Extraction (SCE) and Asymmetric Cluster Extraction (ACE) algorithms are designed to extract significant clusters of features. From the extracted clusters, the final set of opinion feature is derived to formulate robust extractive summaries.

The rest of the paper is organized as follows. Section 2 describes the related works. Section 3 describes the DOFM system. Section 4 describes the automatic cluster extraction mechanisms. Section 5 reports the experimental results and error analysis. Concluding remarks are made in Section 6.

2. Related works

The domain feature retrieval and aspect level sentiment analysis is an important yet challenging issue in the current Internet era. To mine rich domain features, several supervised, unsupervised, and statistical, and deep learning approaches are proposed.

2.1. Supervised approaches

The supervised domain feature retrieval approaches are popular due to their simplicity and scalability; where the models are first trained on a training data set and later evaluated on unseen test data set. The Hidden Markov Model (HMM) is one of the popular supervised methods to train the model by observing the existing process. In Jin, Ho, and Srihari (2009) and Kang, Ahn, and Lee (2018), HMM-based approaches are designed that confirms the applicability of HMM in mining relevant domain feature retrieval. However, HMM poses a few fundamental challenges. First, it is challenging to determine the required number of hidden states to obtain improved performance. Moreover, the required number of hidden states changes from one data set to another, which limits the applicability of HMM-based approaches. Another supervised approach is the conditional random fields (CRFs) that retrieve the domain features by taking the review context into account. In Xia, Yang, Pan, Zhang, and An (2019), conditional random fields (CRFs) based approach is designed to identify domain features in reviews. Contrary to HMM, the CRFs do not have strict independent assumptions, which enable them to accommodate the contextual information. However, CRF-based algorithms are training compute-intensive. For instance, the CRF-based model requires re-training every time new data becomes available. In Wang and Hong (2019) a supervised Hebb rule-based feature selection method is proposed. The Hebb rule-based feature selection is computationally efficient. However, it requires the predefined class information to measure the correlation.

Supervised approaches perform reasonably well, but they have limitations. The foremost limitation is the requirement of a large number of labeled training samples for the model training, which time-consuming, tedious, and labor-intensive task. Moreover, the accuracy of such trained models heavily relies on the quality of the labeling. Moreover, supervised models trained for one domain may not fit to identify domain features for the other domains. In contrast to the supervised approach, DOFM is unsupervised in nature and does not require expensive data labeling. Moreover, by applying an unsupervised technique, DOFM automates the entire domain feature retrieval process.

2.2. Unsupervised approaches

On the contrary, unsupervised approaches do not require labeled training data, which reduces the human intervention and likely improve the performance automating feature retrieval. The topic (Blei, Ng, & Jordan, 2003) and document (Zhao & Mao, 2017) modeling are popular unsupervised approaches to identify domain features from the review sentences. In Blei et al. (2003), the unsupervised topic modeling approach called Latent Dirichlet allocation (LDA) is introduced. The LDA mines the topics and opinion features by learning the latent structures from user reviews. However, in several instances, the extracted features do not correspond to domain features. For cellphone reviews, LDA outputs "delivery" as a valid topic. However, "delivery" is not a cellphone-specific feature. This leads to undermining the effectiveness of LDA. More recently, an unsupervised document modeling approach called Fuzzy Bag-of-Words (FBoW) is introduced in Zhao and Mao (2017) to capture the domain terms. The FBoW performs the cosine similarity measures between word embeddings to mine the domain term. However, FBoW may likely suffer from fundamental shortcoming of cosine similarity measure; where the difference in rating scale of two significantly different reviews results in a high similarity score and degrades the overall performance.

Different from topic and document modeling, unsupervised approaches based on linguistic rules and grammar compositions are also proposed. In Wu, Wu, Wu, Yuan, and Huang (2018) a hybrid linguistic rules-based unsupervised aspect extraction method is designed. The chunk-level linguistic rules are designed to extract nominal phrases and designated as candidate aspects. Later, extracted chunks are used as labeled data to train the gated recurrent units for aspect extraction. However, the performance of the hybrid approach is highly dependent on the accuracy of the extracted nominal phrase and corresponding linguistic rules. In Dragoni, Federici, and Rexha (2019) an unsupervised aspect extraction strategy is designed based on the open information extraction strategy and grammar compositions. The aspect extraction is carried out with the help of compound noun extraction and co-reference resolution. However, designating nouns and pronouns as domain features is misleading and it wrongly captures the non-domain features such as terms "service" and "amazon" from cellphone data set as domain features. In Luo, Huang, and Zhu (2019), a Knowledge Empowered prominent Aspect Extraction (KEAE) approach is designed for the aspect extraction. The KEAE utilizes Probase and WordNet as well as word embeddings for inferring reasonable aspect clusters and extracting prominent aspects. However, the performance of KEAE is highly dependent on Probase and WordNet. Although DOFM is unsupervised, it is not rule-based unlike (Wu et al., 2018). Moreover, DOFM is not built on grammar compositions to tag nouns as domain features. Instead, DOFM analyzes user reviews to identify the statistical patterns based on empirical observation. This improves the ability of DOFM to effectively prune the invalid features unlike (Dragoni et al., 2019).

2.3. Statistical and deep learning approaches

The approaches based on the statistical analysis are helpful to figure out the statistical characteristics of opinion features. The notable contribution is Association Rule Mining (ARM) (Hu & Liu, 2004). The ARM simply designates the frequently appearing noun and noun phrases as domain features. However, in many instances, ARM (Hu & Liu, 2004) outputs frequent yet invalid opinion features such as "service", "amazon" etc. In Zha, Yu, Tang, Wang, and Chua (2014), a Product Aspect Ranking (PAR) framework is formulated by combining the feature sentiments and overall review rating for aspect retrieval. However, PAR (Zha et al., 2014) is built on the assumption of frequent item set mining, which is quite similar to the ARM (Hu & Liu, 2004). Hence, the PAR (Zha et al., 2014) observes only marginal performance gain over ARM (Hu & Liu, 2004). On the contrary, DOFM includes term frequency, grouping-semantics, and distributional statistics, which enhances the performance over ARM (Hu & Liu, 2004) and PAR (Zha et al., 2014).

The statistical analysis based on the domain relevance is also gaining popularity to identify the domain features. In Hai et al. (2014), feature mining using Intrinsic and Extrinsic Domain Relevance (IEDR) is carried out. Two data sets such as domain (DS) data set and domain-independent (DI) data set are employed to list the features with high relevance to the domain data set and at the same time low relevance to the domain-independent data set. The combination of DS and DI data sets is used to effectively prune the overly general features and to improve the results. However, in many instances, IEDR (Hai et al., 2014) prunes the overly general yet valid domain important features. For example, IEDR (Hai et al., 2014) do not consider "size", "price" etc., as cellphone-specific features since they equally appear in other domain-independent data sets such as hotel, culture, etc.

In recent years, domain feature mining using the Deep Learning (DL) techniques (Abas, El-Henawy, Mohamed, & Abdellatif, 2020; Da'u, Salim, Rabiu, & Osman, 2020; Poria, Cambria, & Gelbukh, 2016) and Recurrent Neural Network (RNN) (Wang, Sun, Huang, & Zhu, 2019) are gaining wide acceptability due to its robustness. In Da'u et al. (2020), a weighted Aspect-based Opinion mining using Deep learning (AOD) method is proposed. The AOD extracts the domain features using a DL-based method and fuses them to generate recommendations. In Poria et al. (2016), a seven-layer deep convolutional neural network (DCNN) is proposed to label words aspect or non-aspect word. Later, the DCNN is combined with linguistic patterns to design a sentiment analysis model. In Abas et al. (2020), a deep learning model for fine-grained aspect-based opinion mining is proposed. The proposed work (Abas et al., 2020) trains Google's pre-trained language model on three specific domain corpora for domain adaption and local and global domain features extraction. The DL-based methods show the ability of DL techniques in domain feature retrieval with promising results. However, the DL-based method requires significant computational power to deal with messy and unstructured data. On the contrary, the proposed DOFM requires low computational power due to its unsupervised nature and it incorporates three simple yet robust key observations in domain feature mining. In Wang et al. (2019), an RNN supported aspect-level sentiment capsules model is designed to jointly perform aspect detection and sentiment classification.

3. DOFM: Domain Feature Miner

3.1. System model

Let $\mathbb{R} = \{r_1, r_2, \dots, r_k\}$ be a data set comprised of k reviews. Let $r_i \in \mathbb{R}$ be a set of terms (words) and n_i be the number of distinct terms in r_i after preprocessing and stop words[1] removal. For each r_i, let $W_i = \{w_i^j | j = 1, 2, \dots, n_i\}$ be the set of distinct terms. Consequently, let $W^{\mathbb{D}}$ be a set of distinct terms across the $\mathbb{R}$ and derived as $W^{\mathbb{D}} = \bigcup_{i=1}^{k} W_i$. It is likely that terms appear multiple times in a review such as "battery" in a cellphone review. Let f_i^j be a frequency of w_i^j in r_i, where $f_i^j \geq 1$. Similarly, any term $w_i^j \in W^{\mathbb{D}}$ may also appear multiple times in a data set $\mathbb{R}$. Let F_i^j be a frequency of w_i^j in $\mathbb{R}$, where $F_i^j \geq 1$. The F_i^j is derived as $F_i^j = \sum_{i=1}^{k} f_i^j$.

Let $\mathcal{D}$ be a set of domain features identified by expert annotators from $\mathbb{R}$ and it represents the ground truth. The objective is to mine a sub-set $\mathcal{O}$ of domain features (i.e., terms) from $W^{\mathbb{D}}$ in such a way that $\forall w_i^j \in \mathcal{O}$, the $w_i^j \in \mathcal{D}$, where $\mathcal{O} \subseteq W^{\mathbb{D}}$. The $\mathcal{O}$ represents the set of domain features.

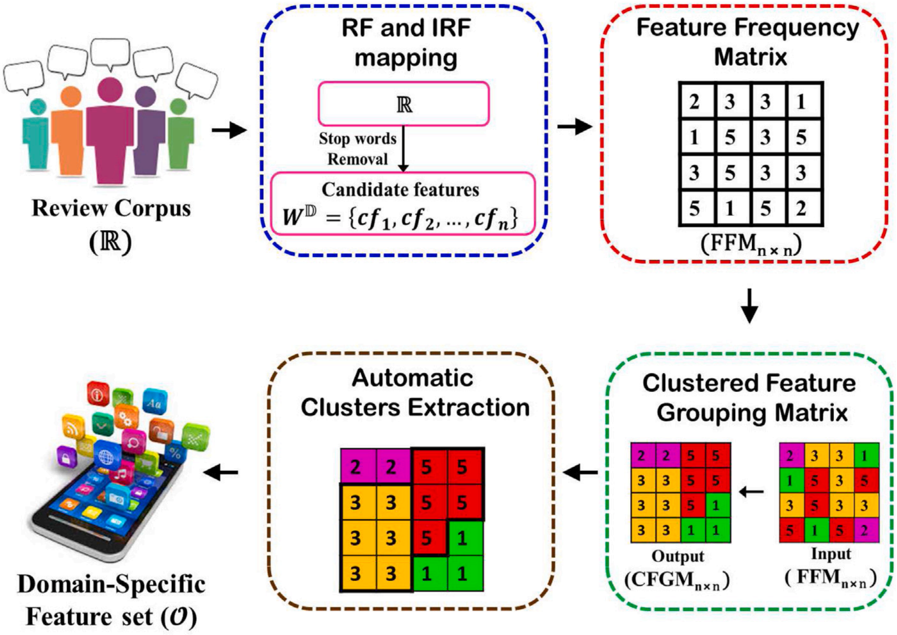
3.2. System objective

Let $\mathcal{D}$ be a set of domain-specific opinion features identified by expert annotators from $\mathbb{R}$. In an opinion feature mining problem, the objective is to mine a sub-set $\mathcal{O}$ of opinion features (i.e., words) from $W^{\mathbb{D}}$, where $\mathcal{O} \subseteq W^{\mathbb{D}}$ in such a way that $\forall w_i^j \in \mathcal{O}$, the $w_i^j \in \mathcal{D}$. The $\mathcal{O}$ represents the set of domain-specific opinion features.

3.3. DOFM framework

The DOFM framework is comprised of four modules as shown in Fig. 2. (1) Review-Feature (RF) mapping and Inverse Review-Feature (IRF) mapping, (2) Feature Frequency Matrix (FFM) formulation, (3) Clustered Feature Grouping Matrix (CFGM) formulation, and (4) Automatic Clusters Extraction. The $\mathbb{R}$ is considered as an input and $W^{\mathbb{D}}$ as candidate features. Let n be the number of candidate features in $\mathbb{R}$ represented as $\{cf_1, cf_2, \dots, cf_n\}$ and organized in an $n \times n$ array called as Feature Frequency

[1] The set of words in a language with least significance. In English, "this", "that", "when", "who", "is", etc., are stop words.



2	3	3	1
1	5	3	5
3	5	3	3
5	1	5	2

2	2	5	5
3	3	5	5
3	3	5	1
3	3	1	1

2	3	3	1
1	5	3	5
3	5	3	3
5	1	5	2

2	2	5	5
3	3	5	5
3	3	5	1
3	3	1	1

Fig. 2. The DOFM framework.

Matrix ($FFM_{n\times n}$). Here, any cell $C_{ij} \in FFM_{n\times n}$ represents the co-appearance frequency of cf_i and cf_j, $i, j = 1, 2, \dots, n$. The $FFM_{n\times n}$ is transformed to $CFGM_{n\times n}$ by permuting columns and rows of a matrix using Bond Energy Algorithm to bring the numerically larger elements together (Arabie & Hubert, 1990). The $CFGM_{n\times n}$ contains symmetric and asymmetric sub-clusters of candidate features, which are difficult to obtain automatically. The customized Symmetric Cluster Extraction (SCE) and Asymmetric Cluster Extraction (ACE) algorithms are designed to extract relevant symmetric and asymmetric clusters, respectively. Finally, symmetric and asymmetric clusters are processed to obtain the domain feature set $\mathcal{O}$. The entire DOFM system process with the example is described in Section 3.4.

3.4. DOFM system with example

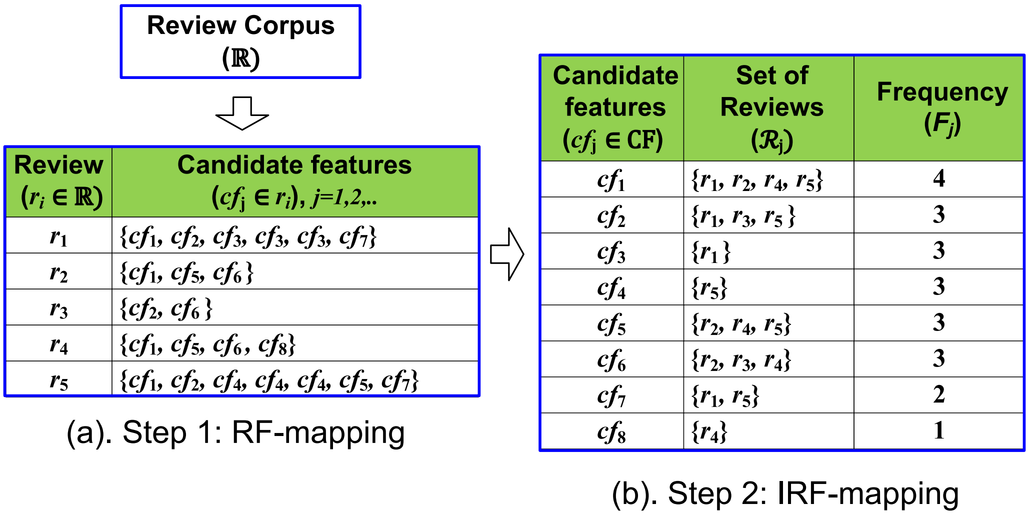
Let $\mathbb{R} = \{r_1, r_2, r_3, r_4, r_5\}$ be a set of cellphone reviews. Each review $r_i \in \mathbb{R}$ is processed to retain distinct terms designated as candidate features. It is to note that candidate features may repeat in a given review. Let us assume that following are the candidate features each review contains after preprocessing. Let r_1 = {camera, display, service, service, service, otg}, r_2 = {camera, speaker, battery}, r_3 = {display, battery}, r_4 = {camera, speaker, battery, order}, and r_5 = {camera, display, delivery, delivery, delivery, speaker, battery}. Consequently, $\mathbb{R}$ contains eight distinct candidate features {camera, display, service, delivery, speaker, battery, otg, order} represented as $W^{\mathbb{D}} = \{cf_1, cf_2, cf_3, cf_4, cf_5, cf_6, cf_7, cf_8\}$, respectively.

3.4.1. RF and IRF mapping

In RF-mapping, each review $r_i \in \mathbb{R}$ is organized in a tabular form by mapping corresponding candidate features as shown in Fig. 3a. In IRF-mapping, the RF-mapping table is scanned and frequency of each candidate feature is obtained across $\mathbb{R}$. For each $cf_j \in W^{\mathbb{D}}$, corresponding frequency F_j and set $\mathscr{R}_j = \{r_i \mid r_i \in \mathbb{R},\ cf_j \in r_i\}$ are obtained as shown in Fig. 3b. Here, $F_j = |\mathscr{R}_j|$. Now, let us explore the frequency-based domain features extraction method. Let cf_j be considered frequent with $F_j > 2$. As shown in Fig. 3b, $\{cf_1, cf_2, \dots, cf_7\}$ qualifies to be domain features with corresponding $F_j \geq 2,\ j = 1, 2, \dots, 7$. However, frequency-based approaches consider frequent yet invalid candidate features as domain features. For instance, cf_3 = "service" is a frequent yet invalid domain feature in a given "cellphone" data set $\mathbb{R}$. Hence, a grouping semantics among the candidate features is explored along with individual frequency to retrieve valid domain features.

3.4.2. Feature Frequency Matrix (FFM)

From an IRF-mapping, FFM is generated to quantify the relationship between all pairs of candidate features (cf_i, cf_j), where $i, j = 1, 2, \dots, 8$. Fig. 4a shows $FFM_{8\times8}$ constructed from the IRF-mapping shown in Fig. 3b. Here, $\forall C_{ij} \in FFM_{8\times8}$, $C_{ij} = |\mathscr{R}_i \cap \mathscr{R}_j|$ represents number of reviews common to cf_i and cf_j. Higher the value of C_{ij}, more frequent the co-appearance of cf_i and cf_j. It is difficult to draw conclusion from $FFM_{8\times8}$ as clusters of candidate features with co-appearance count $C_{ij} \geq 2$ are scattered. Therefore, $FFM_{8\times8}$ is transformed into $CFGM_{8\times8}$ by permuting the rows and columns to gain knowledge.



Review Corpus ($\mathbb{R}$)

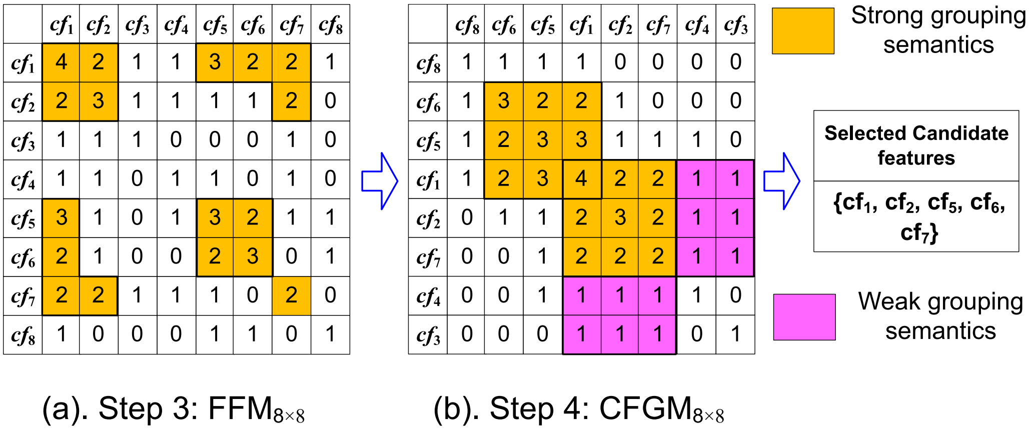
Review ($r_i \in \mathbb{R}$)	Candidate features ($cf_j \in r_i$), $j=1,2,..$
r_1	$\{cf_1, cf_2, cf_3, cf_3, cf_3, cf_7\}$
r_2	$\{cf_1, cf_5, cf_6\}$
r_3	$\{cf_2, cf_6\}$
r_4	$\{cf_1, cf_5, cf_6, cf_8\}$
r_5	$\{cf_1, cf_2, cf_4, cf_4, cf_4, cf_5, cf_7\}$

(a). Step 1: RF-mapping

Candidate features ($cf_j \in$ CF)	Set of Reviews ($\mathcal{R}_j$)	Frequency (F_j)
cf_1	$\{r_1, r_2, r_4, r_5\}$	4
cf_2	$\{r_1, r_3, r_5\}$	3
cf_3	$\{r_1\}$	3
cf_4	$\{r_5\}$	3
cf_5	$\{r_2, r_4, r_5\}$	3
cf_6	$\{r_2, r_3, r_4\}$	3
cf_7	$\{r_1, r_5\}$	2
cf_8	$\{r_4\}$	1

(b). Step 2: IRF-mapping

Fig. 3. Construction of RF-mapping and IRF-mapping from review data set.



	cf_1	cf_2	cf_3	cf_4	cf_5	cf_6	cf_7	cf_8
cf_1	4	2	1	1	3	2	2	1
cf_2	2	3	1	1	1	1	2	0
cf_3	1	1	1	0	0	0	1	0
cf_4	1	1	0	1	1	0	1	0
cf_5	3	1	0	1	3	2	1	1
cf_6	2	1	0	0	2	3	0	1
cf_7	2	2	1	1	1	0	2	0
cf_8	1	0	0	0	1	1	0	1

(a). Step 3: FFM$_{8\times8}$

	cf_8	cf_6	cf_5	cf_1	cf_2	cf_7	cf_4	cf_3
cf_8	1	1	1	1	0	0	0	0
cf_6	1	3	2	2	1	0	0	0
cf_5	1	2	3	3	1	1	1	0
cf_1	1	2	3	4	2	2	1	1
cf_2	0	1	1	2	3	2	1	1
cf_7	0	0	1	2	2	2	1	1
cf_4	0	0	1	1	1	1	1	0
cf_3	0	0	0	1	1	1	0	1

(b). Step 4: CFGM$_{8\times8}$

🟨 Strong grouping semantics

Selected Candidate features
$\{cf_1, cf_2, cf_5, cf_6, cf_7\}$

🟪 Weak grouping semantics

Fig. 4. Construction of $FFM_{8\times8}$ and derivation of $CFGM_{8\times8}$ using BEA. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.4.3. Clustered Feature Grouping Matrix (CFGM)

In CFGM, pair (cf_i, cf_j), where $i, j = 1, 2, \dots, 8$ with $C_{ij} \geq 2$ are grouped together which results in symmetric and asymmetric clusters with strong and weak grouping semantics, respectively. The generated $CFGM_{8\times8}$ is shown in Fig. 4b, which contains two 3×3 symmetric clusters that exhibit strong grouping semantics (in yellow) with $C_{ij} \geq 2$. Similarly, two asymmetric clusters are also generated that exhibit weak grouping semantics (in light pink) with $C_{ij} < 2$. The symmetric clusters show that $\{cf_1, cf_2, cf_5, cf_6, cf_7\}$ not only appear frequently but also co-appear in several reviews in $\mathbb{R}$. On the contrary, asymmetric clusters show that $\{cf_3, cf_4\}$ have no frequent co-appearance and have weak grouping semantics with $\{cf_1, cf_2, cf_7\}$. Consequently, $\{cf_1, cf_2, cf_5, cf_6, cf_7\}$ with corresponding candidate features {camera, display, speaker, battery, otg} are considered as domain features and the rest three $\{cf_3, cf_4, cf_8\}$ with candidate features {service, delivery, order} are ignored.

4. Other parts of DOFM system

4.1. Symmetric Clusters Extraction (SCE)

The process of the SCE is described in Algorithm 1. The $CFGM_{n\times n}$ is an input and the objective is to identify the clusters of cells with $C_{ij} \geq \delta$. Here δ is user-defined threshold. The $CFGM_{n\times n}$ is scanned from left-to-right and top-to-bottom. For each $C_{ij} \in CFGM_{n\times n}$, a temporary matrix $\Pi_{1\times1}$ is initialized with $C_{ij} \geq \delta$. The $\Pi_{1\times1}$ is expanded to $\Pi_{2\times2}$ only if $\forall C_{ij} \in \Pi_{1\times1}$, $C_{ij} \geq \delta$ holds true. Similarly, $\Pi_{2\times2}$ is expanded to $\Pi_{3\times3}$ only if $\forall C_{ij} \in \Pi_{2\times2}$, $C_{ij} \geq \delta$ holds true. The temporary matrix expansion process terminate at $\Pi_{k\times k}$ only if $\exists C_{ij} \in \Pi_{k\times k}$ with $C_{ij} < \delta$.

Algorithm 1: Symmetric Clusters Extraction (SCE).

```
Input: $CFGM_{n\times n}$.
Output: Symmetric clusters $SC$.
Notations:
$\delta$ = Predefined threshold,
$C_{ij}$ = Value of $i$th row and $j$th column of $CFGM_{n\times n}$,
$N\_Rows(X_{p\times q})$ = Number of rows in matrix $X_{p\times q}$.
Initialize $\delta$ ;
Initialize $SC = null$ ;
Scan $CFGM_{n\times n}$ left-right and top-bottom ;
foreach $C_{ij} \geq \delta$ do
    Initialize temporary cluster matrix $\Pi_{1\times 1}$ ;
    Insert cell $C_{ij}$ into $\Pi_{1\times 1}$ ;
    Assign $k = N\_Rows(\Pi_{1\times 1})$ ;
    while $\forall C_{ij} \in \Pi_{k\times k} \geq \delta$ do
        Expand $\Pi_{k\times k}$ to $\Pi_{(k+1)\times(k+1)}$ ;
        $k = k + 1$ ;
    end
    $SC = SC \cup \Pi_{(k-1)\times(k-1)}$ ;
end
Discard all proper subset clusters from $SC$ ;
return $SC$ ;
```

Fig. 5 shows an example of SCE for a $CFGM_{8\times 8}$ constructed in Section 3.4.3. The $CFGM_{8\times 8}$ is scanned from left-to-right and top-to-bottom to identify the cell $C_{ij} \geq \delta$. Let $\delta = 2$. As shown in Fig. 5a, $C_{22} \in CFGM_{8\times 8}$ and $C_{22} \geq \delta$. Therefore, temporary cluster expansion $\Pi_{1\times 1}$ starts from C_{22} shown in yellow. As shown in Fig. 5b, the $\Pi_{1\times 1}$ incrementally expands to $\Pi_{2\times 2}$ as $\forall C_{ij} \in \Pi_{2\times 2}$, $C_{ij} \geq \delta$. Similarly, $\Pi_{2\times 2}$ incrementally expands to $\Pi_{3\times 3}$ as $\forall C_{ij} \in \Pi_{3\times 3}$, $C_{ij} \geq \delta$ shown in Fig. 5c. The temporary cluster expansion process terminates at $\Pi_{4\times 4}$, as $\exists C_{ij} \in \Pi_{4\times 4}$ with $C_{ij} < \delta$. For instance, $C_{25} \in \Pi_{4\times 4} = 1$, and $C_{25} < (\delta = 2)$. Therefore, a symmetric cluster $\Pi_{3\times 3}$ is considered with respect to C_{22}. The aforementioned process repeats $\forall C_{ij} \in CFGM_{8\times 8}$ with $C_{ij} \geq \delta$. In Fig. 5a, there are 17 cells with $C_{ij} \geq \delta = 2$ and therefore a set SC of 17 symmetric clusters will be generated. However, all subset clusters from SC are discarded to retain large superset clusters. For a given example, SCE outputs two large symmetric clusters (in yellow) as shown in Fig. 4b.

Theorem 1. *The time-complexity for SCE from $CFGM_{n\times n}$ is bounded by $\mathcal{O}(Cn^2)$.*

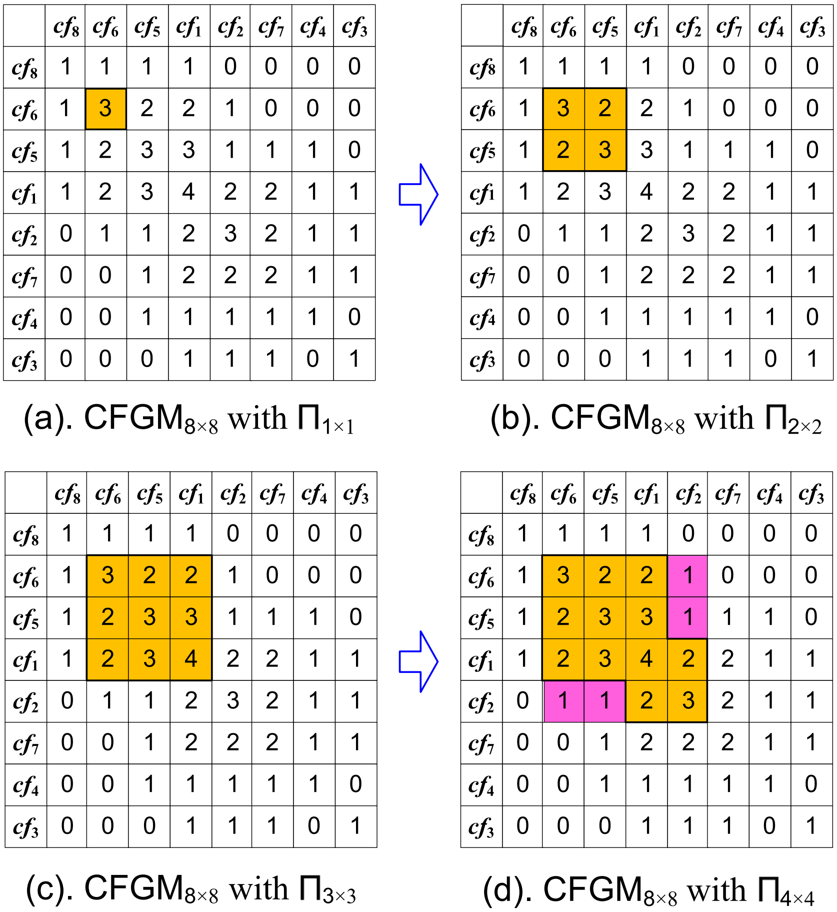
Proof. The SCE is an iterative step by step expansion process from $\Pi_{1\times 1}$ to $\Pi_{k\times k}$ until $\exists C_{ij} \in \Pi_{k+1\times k+1}$ with $C_{ij} < \delta$.

- The SCE contains two sub-processes such as a matrix scanning and iterative temporary cluster formation.
- The unit operation in a matrix scanning is a "search" and in iterative temporary cluster formation is a "comparison". The unit operations "search" and "comparison" are considered as basis for the time complexity calculation.
- *Search complexity:* In a $CFGM_{n\times n}$, a matrix scanning takes $n \times n$ search operations. Let C be a constant that represents $n \times n$ matrix cells. Therefore, worst-case search time complexity is $\mathcal{O}(C)$.
- *Temporary cluster formation complexity:* On the other hand, SCE results in an iterative temporary cluster formation of size $\Pi_{k\times k}$ in the worst case scenario, where $k = n$.
- $\forall C_{ij} \in CFGM_{n\times n}$, the temporary cluster formation of size $\Pi_{k\times k}$ takes maximum $\mathcal{O}(k^2)$ number of comparisons. For $k = n$, it takes $\mathcal{O}(n^2)$ comparisons.
- The aforementioned process repeats for each of the C number of cells, which results in $\mathcal{O}(Cn^2)$. Hence the worst-case time complexity of SCE is bounded by $\mathcal{O}(Cn^2)$. □

4.2. Asymmetric Clusters Extraction (ACE)

The ACE algorithm identifies asymmetric clusters with $C_{ij} \geq \delta$. The detailed process of ACE is described in Algorithm 2. The ACE takes $CFGM_{n\times n}$ as input and outputs a set AC of asymmetric clusters. Similar to SCE, ACE also chooses the threshold δ using heuristic methods described in Section 4.3. Unlike SCE, ACE first expands column-wise followed by row-wise to formulate $\Pi_{n\times m}$ in such a way that $n \neq m$, and $\forall C_{ij} \in \Pi_{n\times m}$, the $C_{ij} \geq \delta$.

In order to explain ACE, few cell values of Fig. 5 is changed to prepare new $CFGM_{8\times 8}$ as shown in Fig. 6. The $CFGM_{8\times 8}$ is scanned from left-to-right and top-to-bottom to identify the cell $C_{ij} \geq \delta$. Let $\delta = 2$. As shown in Fig. 6a, $C_{22} \in CFGM_{8\times 8}$ and $C_{22} \geq \delta$. Therefore, temporary cluster expansion $\Pi_{1\times 1}$ starts from C_{22} shown in yellow. Contrary to SCE, ACE first expands column-wise followed by row-wise. As shown in Fig. 6b, the $\Pi_{1\times 1}$ incrementally expands column-wise to $\Pi_{1\times 2}$, $\Pi_{1\times 3}$, and $\Pi_{1\times 4}$ as



	cf_8	cf_6	cf_5	cf_1	cf_2	cf_7	cf_4	cf_3
cf_8	1	1	1	1	0	0	0	0
cf_6	1	3	2	2	1	0	0	0
cf_5	1	2	3	3	1	1	1	0
cf_1	1	2	3	4	2	2	1	1
cf_2	0	1	1	2	3	2	1	1
cf_7	0	0	1	2	2	2	1	1
cf_4	0	0	1	1	1	1	1	0
cf_3	0	0	0	1	1	1	0	1

(a). CFGM$_{8\times8}$ with $\Pi_{1\times1}$

	cf_8	cf_6	cf_5	cf_1	cf_2	cf_7	cf_4	cf_3
cf_8	1	1	1	1	0	0	0	0
cf_6	1	3	2	2	1	0	0	0
cf_5	1	2	3	3	1	1	1	0
cf_1	1	2	3	4	2	2	1	1
cf_2	0	1	1	2	3	2	1	1
cf_7	0	0	1	2	2	2	1	1
cf_4	0	0	1	1	1	1	1	0
cf_3	0	0	0	1	1	1	0	1

(b). CFGM$_{8\times8}$ with $\Pi_{2\times2}$

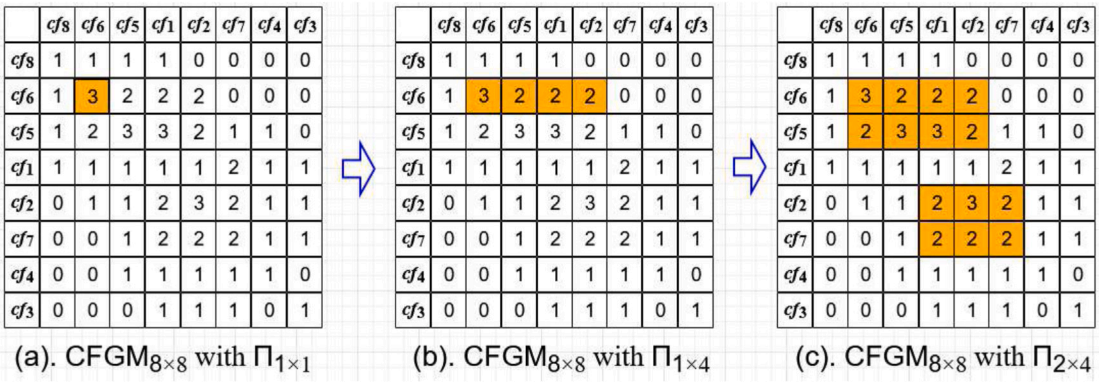
	cf_8	cf_6	cf_5	cf_1	cf_2	cf_7	cf_4	cf_3
cf_8	1	1	1	1	0	0	0	0
cf_6	1	3	2	2	1	0	0	0
cf_5	1	2	3	3	1	1	1	0
cf_1	1	2	3	4	2	2	1	1
cf_2	0	1	1	2	3	2	1	1
cf_7	0	0	1	2	2	2	1	1
cf_4	0	0	1	1	1	1	1	0
cf_3	0	0	0	1	1	1	0	1

(c). CFGM$_{8\times8}$ with $\Pi_{3\times3}$

	cf_8	cf_6	cf_5	cf_1	cf_2	cf_7	cf_4	cf_3
cf_8	1	1	1	1	0	0	0	0
cf_6	1	3	2	2	1	0	0	0
cf_5	1	2	3	3	1	1	1	0
cf_1	1	2	3	4	2	2	1	1
cf_2	0	1	1	2	3	2	1	1
cf_7	0	0	1	2	2	2	1	1
cf_4	0	0	1	1	1	1	1	0
cf_3	0	0	0	1	1	1	0	1

(d). CFGM$_{8\times8}$ with $\Pi_{4\times4}$

Fig. 5. Example of SCE from $\Pi_{1\times1}$ to $\Pi_{4\times4}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



	cf_8	cf_6	cf_5	cf_1	cf_2	cf_7	cf_4	cf_3
cf_8	1	1	1	1	0	0	0	0
cf_6	1	3	2	2	2	0	0	0
cf_5	1	2	3	3	2	1	1	0
cf_1	1	1	1	1	1	2	1	1
cf_2	0	1	1	2	3	2	1	1
cf_7	0	0	1	2	2	2	1	1
cf_4	0	0	1	1	1	1	1	0
cf_3	0	0	0	1	1	1	0	1

(a). CFGM$_{8\times8}$ with $\Pi_{1\times1}$

	cf_8	cf_6	cf_5	cf_1	cf_2	cf_7	cf_4	cf_3
cf_8	1	1	1	1	0	0	0	0
cf_6	1	3	2	2	2	0	0	0
cf_5	1	2	3	3	2	1	1	0
cf_1	1	1	1	1	1	2	1	1
cf_2	0	1	1	2	3	2	1	1
cf_7	0	0	1	2	2	2	1	1
cf_4	0	0	1	1	1	1	1	0
cf_3	0	0	0	1	1	1	0	1

(b). CFGM$_{8\times8}$ with $\Pi_{1\times4}$

	cf_8	cf_6	cf_5	cf_1	cf_2	cf_7	cf_4	cf_3
cf_8	1	1	1	1	0	0	0	0
cf_6	1	3	2	2	2	0	0	0
cf_5	1	2	3	3	2	1	1	0
cf_1	1	1	1	1	1	2	1	1
cf_2	0	1	1	2	3	2	1	1
cf_7	0	0	1	2	2	2	1	1
cf_4	0	0	1	1	1	1	1	0
cf_3	0	0	0	1	1	1	0	1

(c). CFGM$_{8\times8}$ with $\Pi_{2\times4}$

Fig. 6. Example of ACE from $\Pi_{1\times1}$ to $\Pi_{2\times4}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$\forall C_{ij} \in \Pi_{1\times k}$, $C_{ij} \geq \delta$ with $k \in \{2,3,4\}$. The column-wise temporary cluster expansion process terminates at $\Pi_{1\times4}$, as $\exists C_{ij} \in \Pi_{1\times5}$ with $C_{ij} < \delta$. Similarly, $\Pi_{1\times4}$ incrementally expands to $\Pi_{2\times4}$ in a row-wise as $\forall C_{ij} \in \Pi_{2\times4}$, $C_{ij} \geq \delta$ shown in Fig. 6c. The row-wise temporary cluster expansion process terminates at $\Pi_{2\times4}$, as $\exists C_{ij} \in \Pi_{3\times4}$ with $C_{ij} < \delta$. In Fig. 6a, there are 15 cells with $C_{ij} \geq \delta = 2$

and therefore a set AC of 15 Asymmetric clusters will be generated. However, all subset clusters from AC are discarded to retain large superset clusters. For a given example, ACE outputs two large Asymmetric clusters (in yellow) as shown in Fig. 6c.

Algorithm 2: Asymmetric Clusters Extraction (ACE).

```
Input: $CFGM_{n\times n}$.
Output: Asymmetric clusters ($AC$).
Notations:
$\delta$ = Predefined threshold,
$C_{ij}$ = Value of $i$th row and $j$th column of $CFGM_{n\times n}$,
$N\_Rows(X_{p\times q})$ = Number of rows in matrix $X_{p\times q}$,
$N\_Cols(X_{p\times q})$ = Number of columns in matrix $X_{p\times q}$.
Initialize value of $\delta$ ;
Initialize $AC = null$ ;
Scan $CFGM$ left-right and top-bottom ;
foreach $C_{ij} \geq \delta$, where $C_{ij} \in CFGM_{n\times n}$ do
    Initialize temporary matrix $\Pi_{1\times 1}$ ;
    Insert cell $C_{ij}$ into $\Pi_{1\times 1}$ ;
    Assign $k = N\_Rows(\Pi_{1\times 1})$ ;
    Assign $l = N\_Cols(\Pi_{1\times 1})$ ;
    $m = l + 1$ ;
    while $C_{im} \geq \delta$, where $C_{im} \in CFGM_{n\times n}$ do
        Expand $\Pi_{k\times l}$ to $\Pi_{k\times m}$ ;
        Insert cell $C_{im}$ into $\Pi_{k\times m}$ ;
        $l = m$ ;
        $m = m + 1$ ;
    end
    foreach $C_{iy} \in \Pi_{i\times(m-1)}$, where $y = l$ to $m-1$ do
        $n = i + 1$ ;
        while $C_{ny} \geq \delta$, where $C_{ny} \in CFGM_{n\times n}$ do
            Expand $\Pi_{i\times(m-1)}$ to $\Pi_{n\times(m-1)}$ ;
            Insert cell $C_{ny}$ into $\Pi_{n\times(m-1)}$ ;
            $n = n + 1$ ;
        end
    end
    $AC = AC \cup \Pi_{(n-1)\times(m-1)}$ ;
end
Discard all proper subset clusters from $AC$ ;
return $AC$ ;
```

Theorem 2. *The time-complexity for ACE from $CFGM_{n\times n}$ is bounded by $\mathcal{O}(Cn^2)$.*

Proof. The ACE is also an iterative step by step expansion process from $\Pi_{1\times 1}$ to $\Pi_{n\times m}$ until $\exists C_{ij} \in \Pi_{n+1\times m+1}$ with $C_{ij} < \delta$.

- The ACE time-complexity can be proved similar to SCE with two sub-processes such as a matrix scanning and iterative temporary cluster formation with unit operations "search" and "comparison", respectively.
- *Search complexity:* The ACE takes constant $n \times n$ number of search operations to scan $CFGM_{n\times n}$ matrix, which results in $\mathcal{O}(C)$ for $C = n \times n$.
- *Temporary cluster formation complexity:* Contrary to SCE, the maximum size of a temporary matrix in ACE can be $\Pi_{n\times(n-1)}$ or $\Pi_{(n-1)\times n}$, which requires at most $n^2 - n$ number of comparisons represented as $\mathcal{O}(n^2)$.
- $\forall C_{ij} \in CFGM_{n\times n}$, the temporary cluster formation of size $\Pi_{k\times k}$ takes maximum $\mathcal{O}(k^2)$ number of comparisons. For $k = n$, it takes $\mathcal{O}(n^2)$ comparisons.
- The temporary matrix formation process repeats for each of the C cells of $CFGM_{n\times n}$. Hence, the worst case time complexity can be represented as $\mathcal{O}(Cn^2)$. □

4.3. Heuristic approaches to decide threshold δ

The δ can be decided experimentally. However, it is a tedious and time-consuming process. Moreover, the δ obtained for one data set may not be applicable to another. Hence, heuristics methods are employed to obtain δ considering the distribution of candidate features in $\mathbb{R}$. If candidate features follow the Gaussian Distribution, then mean can be used as a threshold represented as δ_μ. The

Table 1
Statistical information of domain review data sets.

Data set	# of reviews	Avg # of sentences	Avg # of words	Avg # of candidate features
camera	9000	4.90	38.90	27.69
laptop	9000	17.72	155.63	119.20
cellphone	9000	9.27	55.72	34.26
tablet	9000	13.18	102.09	70.15
television	9000	20.81	176.45	104.09
hotel	9000	7	41.44	33.83

δ_μ is calculated from the value of cells in the main diagonal of $CFGM_{n\times n}$ as given in Eq. (1).

$$\delta_\mu = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} C_{ij}}{n}, \ \forall i = j \tag{1}$$

For skewed distribution of candidate features, median acts as a suitable measure to calculate threshold represented as δ_M. The δ_M is calculated by arranging the candidate features in ascending order of their frequency in $\mathbb{R}$ followed by selecting a middle value of the ordered list.

The another heuristic method is based on the histogram of values present in the main diagonal of $CFGM_{n\times n}$. The histogram reveals the proportional distribution of individual values. The $CFGM_{8\times 8}$ shown in Fig. 4b has four possible values such as 1, 2, 3 and 4 in main diagonal with proportional distribution of $\frac{3}{8}$, $\frac{1}{8}$, $\frac{3}{8}$ and $\frac{1}{8}$, respectively. To estimate the threshold, values from maximum to minimum are selected and their proportional distribution is added until it crosses half of the distribution. The added proportional distribution of 4 and 3 is $(\frac{1}{8}+\frac{3}{8}) = \frac{4}{8}$, which is still not greater than $\frac{4}{8}$. Hence, proportional distribution of 2 is added, which results in total proportional distribution of $(\frac{1}{8}+\frac{3}{8}+\frac{1}{8}) = \frac{5}{8} > \frac{4}{8}$ The minimum value out of 4, 3, and 2 is chosen as representative histogram based threshold represented as δ_H.

5. Experimental study

5.1. Description of data sets

To validate the effectiveness of the DOFM in terms of domain feature mining, we evaluate the DOFM using Amazon and TripAdvisor data sets acquired from the University of Illinois at Urbana–Champaign (Wang et al., 2011), which is widely used in related works (Pham & Le, 2018; Zhang, Barzilay, & Jaakkola, 2017; Zhao & Mao, 2017). The data sets consist of review sentences from six different domains such as “camera”, “laptop”, “cellphone”, “tablet”, “television”, and “hotel”. Each domain is consists of 9000 randomly chosen reviews, which together form a data set of size 54000 reviews. The statistical detail of the data sets is shown in Table 1. The data sets are comprised of a set of JSON files and each JSON file is comprised of several reviews that store information on attributes “title”, “review id”, “reviewer”, “date”, “overall rating”, and “user review” in a key–value format. For domain feature analysis, only the “user reviews” are considered and other attributes are discarded. The television reviews are detailed with the highest average number of sentences and words per review; whereas camera reviews are shallow with the lowest average number of sentences and words per review. The average number of candidate features represents the words other than stop words.

5.2. Experimental details

Each data set is pre-processed on MATLAB R2017a to remove the bias and to keep the content coverage uniform as follow. (1) Reviews with less than 05 words are discarded as they are less informative. (2) All the terms are changed to lower cases. (3) Special symbols and punctuations are removed. (4) Stop words are removed using the Terrier information retrieval platform[2] to obtain candidate features. The domain feature labeling is carried out for all six domains with the help of two domain experts. Each review sentence is analyzed by both experts to label the domain features. For example, in a sentence, “The cellphone *battery* is good, but the *camera* is bad”, both experts are asked to label domain features “battery” and “camera”. The conflict between the experts is resolved by engaging a third expert. The outcome of expert labeling of randomly chosen 1000 reviews is described as follows. The inter-annotator reliability is 0.77, which is measured using Cohen’s kappa coefficient (Sim & Wright, 2005). The kappa coefficient between 0.61 to 0.8 is considered a substantial agreement.

[2] http://terrier.org/.

Table 2
Comparison of sample output opinion feature extracted for cellphone data set.

Annotators	ARM	DOFM	IEDR	PAR	AORF	KEAE
camera	camera	camera	camera	camera	camera	camera
battery	battery	battery	battery	battery	battery	battery
screen	screen	screen	screen	screen	screen	screen
sound	sound	sound	sound	sound	sound	sound
size	size	size	×	size	size	size
price	price	price	×	price	price	price
×	**service**	×	**service**	**service**	**service**	**service**
×	**amazon**	×	×	**amazon**	**amazon**	×

5.3. Quality assessment of domain features

The DOFM extracted domain features are compared with those extracted by state-of-the-art approaches and the expert annotators.

The description of the rival approaches is as follow: (a) Expert annotators, where domain features are manually extracted by experts; (b) IEDR (Hai et al., 2014), where domain features are extracted using domain-dependent and domain-independent data sets; (c) ARM (Hu & Liu, 2004), where frequent nouns are considered domain features; (d) PAR (Zha et al., 2014), where domain feature are extracted and ranked using probabilistic approach; (e) Aspect-based Opinion Ranking Framework (AORF) (Kumar & Abirami, 2018), where nouns and pronouns are considered as domain features; (f) Knowledge Empowered prominent Aspect Extraction (KEAE) (Luo et al., 2019), where sources such as Probase and WordNet is used for the aspect extraction.

The sample output produced by the approaches for "cellphone" data set is shown in Table 2. The expert annotated domain features are considered ground truth to evaluate the approaches. The "cellphone" reviews are obtained from the Amazon and it is likely that reviewers may rate services offered by amazon as well. As shown in Table 2, "service" and "amazon" are non-domain features. However, ARM (Hu & Liu, 2004) and AORF (Kumar & Abirami, 2018) captures them as domain features due to the high-frequency count. The PAR (Zha et al., 2014) assumes that domain features likely appear in several reviews, which is similar assumption to that of ARM (Hu & Liu, 2004). Hence, the outcome of PAR (Zha et al., 2014) is similar to that of ARM (Hu & Liu, 2004) and AORF (Kumar & Abirami, 2018). On the contrary, IEDR (Hai et al., 2014) and KEAE (Luo et al., 2019) successfully avoid the non-domain feature "amazon", but retains the "service", when "cellphone" data set is analyzed. However, IEDR (Hai et al., 2014) also excludes the generic features "size" and "price", which equally appear in domain-dependent data set "cellphone" and domain-independent data set "hotel". The potential reason for improved performance of KEAE is its effective strategy of narrowing the domain feature space utilizing the Probase and Wordnet.

On the contrary, output of the DOFM is similar to the one reported by expert annotators. The non-domain features "amazon" and "service" are frequent, but their skewed distribution across "cellphone" data set results into their lower co-appearance count. This leads to their rejection as domain features in the DOFM.

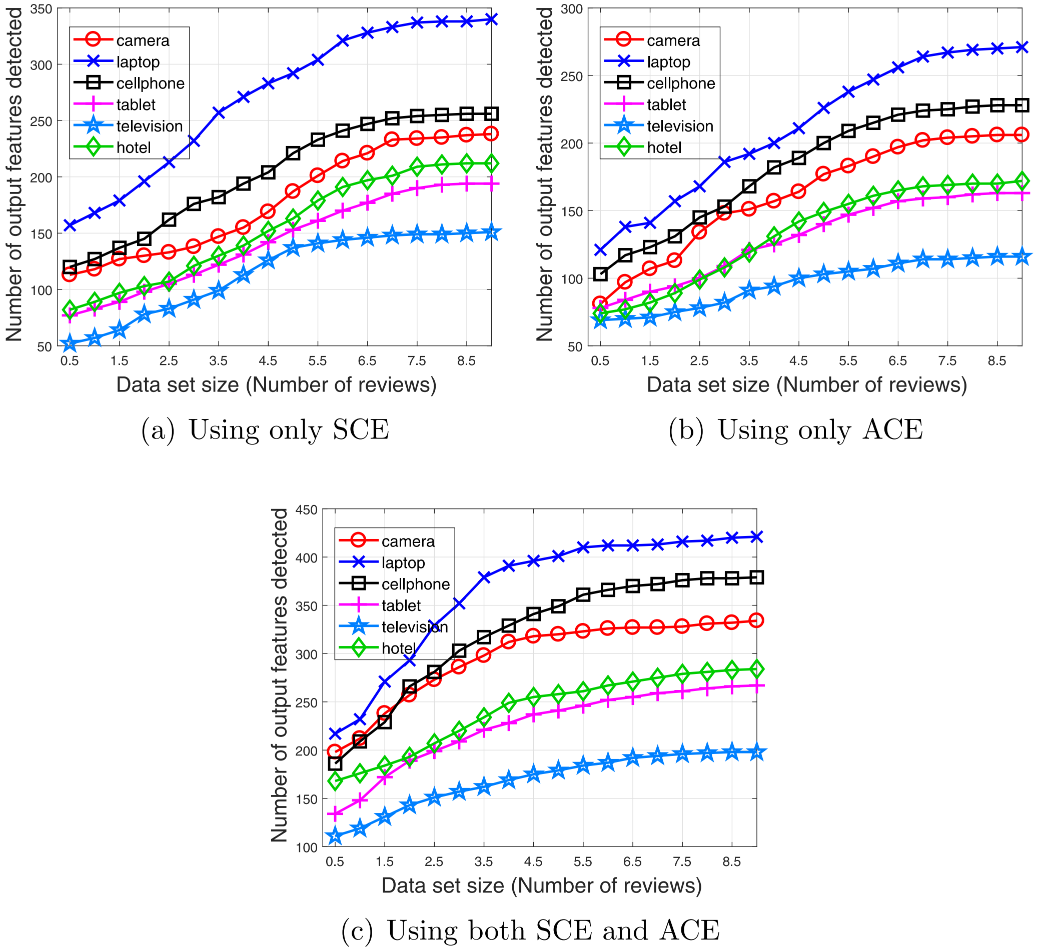
5.4. Impact of data set size

An empirical analysis is performed to ascertain the impact of data set and review size on the domain feature retrieval system. Fig. 7 reports the number of domain features extracted with reference to data set size. Three separate experiments are carried out such as (a) using only SCE, (b) using only ACE, and (c) using both SCE and ACE with δ_μ threshold.

The experimental outcome concerning the data set size is described as follow. Fig. 7 confirms that the number of domain features increases with the data set size and saturates towards the end. This infers that beyond a sufficient number of reviews, the probability of mining unseen domain feature is negligible. It is observed the SCE is more successful than ACE in mining domain features given a data set size. However, the SCE and ACE together mine more number of domain features compared to their individual use. The results of Fig. 7 are in line with the statistical analysis of data sets reported in Table 1. The DOFM mines the highest number of domain features i.e. 421 from "laptop" data set in combined use of SCE and ACE, which also contains the highest number of candidate features as reported in Table 1. On the contrary, DOFM mines only 198 domain features from "television" data set, which contradicts to its statistical analysis reported in Table 1. The "television" data set reviews are detailed and contain more number of candidate features as reported in Table 1. However, our analysis reveals that most candidate features do not qualify to be domain features considering their lower co-appearance count.

5.5. Impact of review size

An empirical analysis is performed to ascertain the impact of the review size. The reviews are manually classified into small, medium, and large categories based on the size. Fig. 8 reports the number of domain features extracted across the three categories for each data set. The experimental outcome concerning the review size is described as follow. Fig. 8 reports that there is a linear relationship between the domain feature extraction and review size. With the increase in review size, the probability of mining more number features also increases proportionally. This linear relationship is obvious due to human psychology. The large-sized user reviews are likely to be more detailed in nature, which likely increases the probability of more number of feature inclusion by the users.



(a) Using only SCE

(b) Using only ACE

(c) Using both SCE and ACE

Fig. 7. Number of output opinion features on different data set size, (a) Using only SCE, (b) Using only ACE, (c) Using both SCE and ACE.

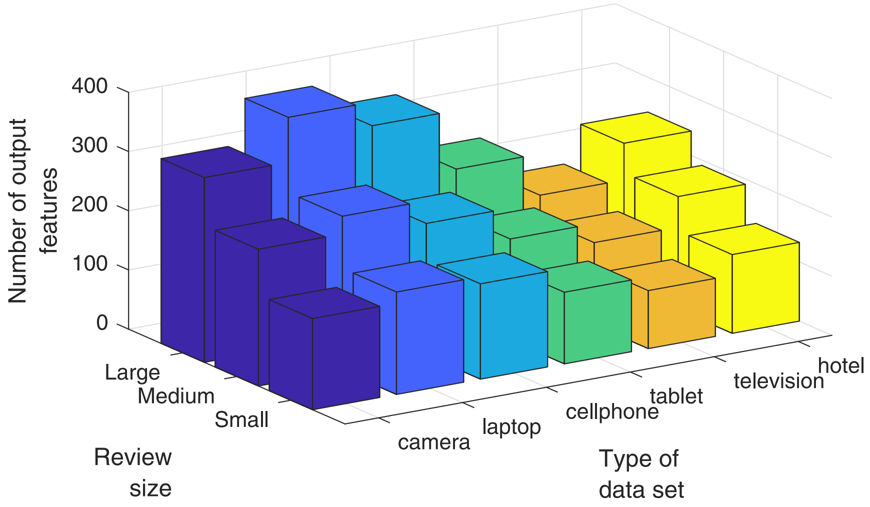
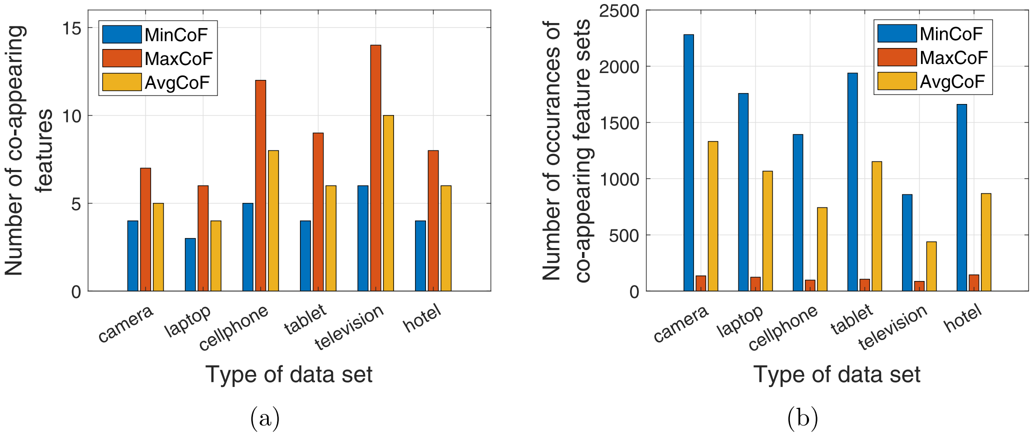


Fig. 8. Number of output opinion features on different sizes of reviews.

Another empirical study is performed to address the question, what is the maximum and minimum size of co-appearing features set. In other words, the study intends to know the number of features co-appear across the 9000 reviews and their co-appearance



(a) (b)

Fig. 9. (a) The maximum, minimum, and average size of co-appearing feature set, (b) The # of occurrences of maximum and minimum number of co-appearing features.

Table 3
List of Top-10 opinion features.

camera	laptop	cellphone	tablets	television	hotel
lens	processor	battery	screen	sound	service
zoom	ram	camera	apps	price	location
resolution	screen	screen	price	apps	cleanliness
screen	drive	sound	size	internet	staff
sound	design	speaker	battery	screen	parking
price	os	volume	wifi	remote	breakfast
battery	macbook	price	camera	wifi	internet
size	software	design	display	hdmi	bathroom
memory	battery	bluetooth	speakers	size	price
digital	cd	memory	4g	lcd	size

frequency count. This helps us to identify the opinion features that are top-rated, highly influential, and preferred by many users to review the product or service. In Fig. 9a, the minimum, maximum, and average size of the co-appearing feature set is reported for each data set represented as $MinCoF$, $MaxCoF$, and $AvgCoF$, respectively. On the contrary, Fig. 9b reports the corresponding frequency count, i.e., # of occurrences, of $MinCoF$, $MaxCoF$, and $AvgCoF$.

The experimental outcome for the co-appearing feature set is described as follow. The largest co-appearing feature set with $MaxCoF = 14$ opinion features is observed for "television" data set, followed by "cellphone" with $MaxCoF = 12$, "tablet" with $MaxCoF = 09$, "hotel" with $MaxCoF = 08$, "camera" with $MaxCoF = 07$, and "laptop" with $MaxCoF = 06$. A similar trend is observed for $MinCoF$ and $AvgCoF$ across all of the data sets, except for "tablet" and "hotel" data sets, where the value of $MinCoF$ and $AvgCoF$ are observed the same. For each data set, the $MinCoF$ results with high frequency count, followed by $AvgCoF$ and $MaxCoF$. This trend reveals the fact that the larger the size of the co-appearing feature set, the lesser the corresponding frequency count.

From the aforementioned analysis, Top-10 opinion features are identified for each data set and the same are reported in Table 3. Table 3 provides the deeper insight and shows the important opinion features per data set. For example, "lens" is found highly important feature among users reviewing camera products; whereas "service" is found an important feature for hotel reviewers. It is observed that few features are found common across different products. For instance, "battery" appears in "camera", "laptop", "cellphone", and "tablet" data sets. On the other hand, "battery" is found common to "tablet", and "cellphone" data sets.

5.6. Quantitative evaluation of the DOFM

The comprehensive evaluation of the DOFM is performed using various quality evaluation metrics such as Precision, Recall, and F-score as given in Eqs. (2), (3), and (4), respectively. Here, tp = true positive, fp = false positive, fn = false negative.

$$Precision = \frac{tp}{tp + fp} \tag{2}$$

$$Recall = \frac{tp}{tp + fn} \tag{3}$$

$$F - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

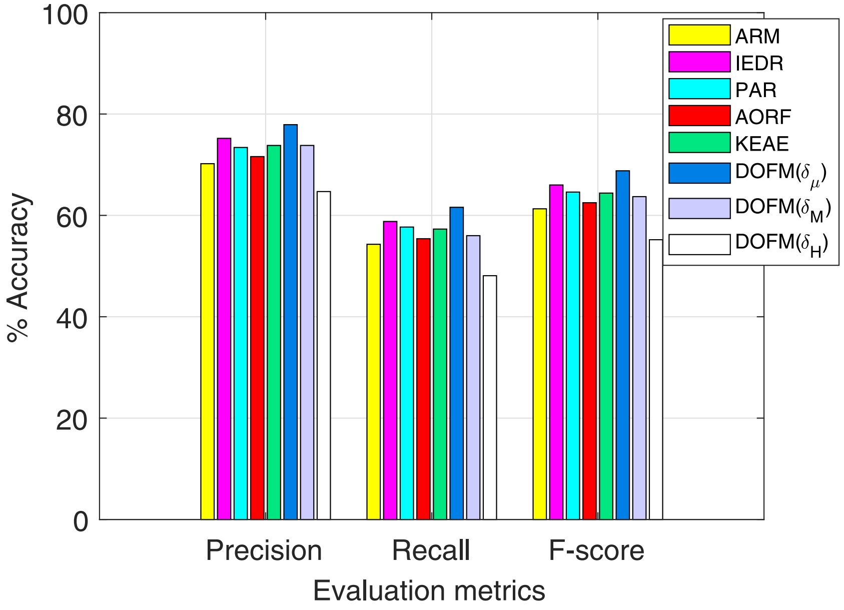


Fig. 10. Results of opinion feature set mining for camera data set. The results are statistically significant with *T-Test*, *P-Values* < 0.05.

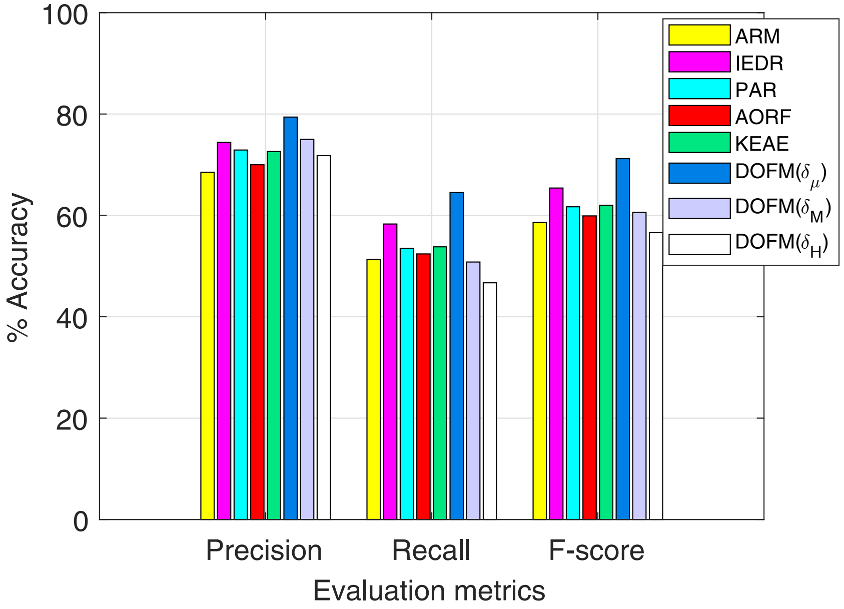


Fig. 11. Results of opinion feature set mining for laptop data set. The results are statistically significant with *T-Test*, *P-Values* < 0.05.

For each of the six data sets, the evaluation results are reported in Figs. 10, 11, 12, 13, 14, and 15 using the 1000 sample reviews. The DOFM performance evaluation is carried out for all heuristic approaches with thresholds DOFM (δ_{μ}), DOFM (δ_{M}), and DOFM (δ_{H}).

The DOFM (δ_{μ}) performs consistently well for all the data sets except the “cellphone” data set across all evaluation metrics with the significant increase in Precision, Recall, and F-score. On the contrary, DOFM (δ_{μ}) provides better results on two evaluation metrics for the “hotel” data set, i.e., Recall and F-score with values 65.33% and 72.04%, respectively; whereas DOFM (δ_{M}) provides the improved result on Precision with value 81.41%.

Among the existing methods, the ARM (Hu & Liu, 2004), AORF (Kumar & Abirami, 2018), and PAR (Zha et al., 2014) report the nearly identical performance in term of Precision, Recall, and F-score for all the data sets. The potential reason behind the reduced performance is their inability to effectively distinguish between frequently appearing domain features to that of frequently appearing non-domain features. For example, ARM (Hu & Liu, 2004), AORF (Kumar & Abirami, 2018), and PAR (Zha et al., 2014)

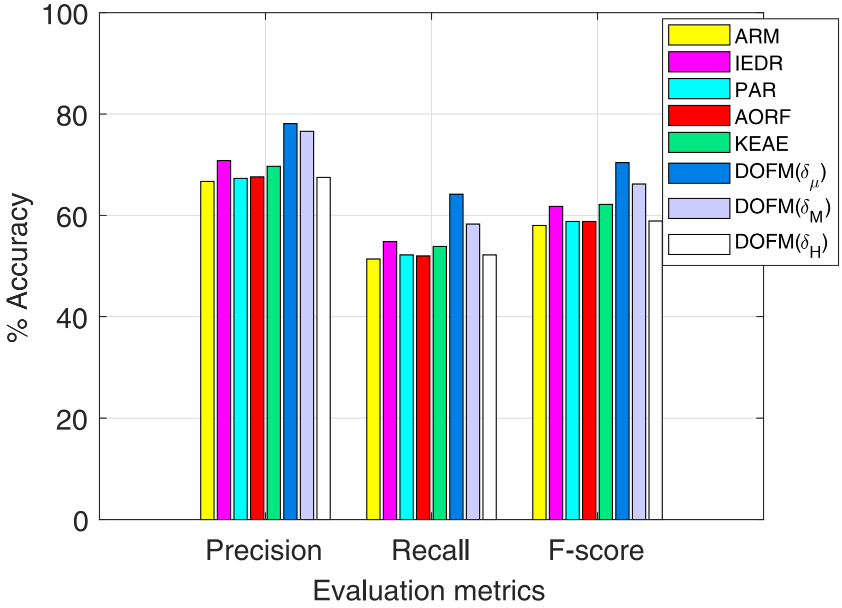


Fig. 12. Results of opinion feature set mining for cellphone data set. The results are statistically significant with *T-Test*, *P-Values* < 0.05.

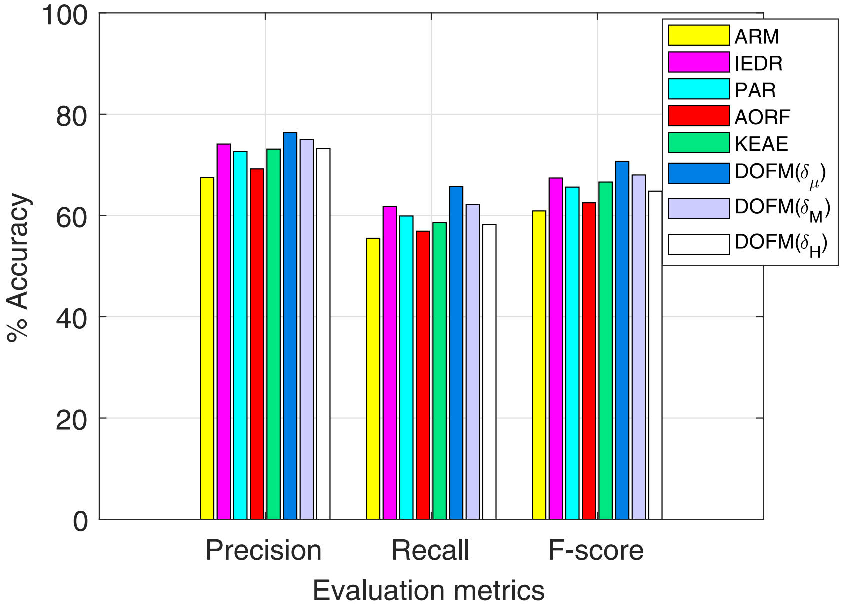


Fig. 13. Results of opinion feature set mining for tablet data set. The results are statistically significant with *T-Test*, *P-Values* < 0.05.

extract "service" and "amazon" as domain features due to their frequent appearance, which results in their reduced performance. However, "service" and "amazon" are non-domain features and must not be considered. It is observed that the IEDR (Hai et al., 2014) and KEAE (Luo et al., 2019) perform reasonably well compared to ARM (Hu & Liu, 2004), AORF (Kumar & Abirami, 2018), and PAR (Zha et al., 2014). The possible reason behind the improved performance of IEDR (Hai et al., 2014) lies in its ability to identify and discard the non-domain features up to some extent. For example, IEDR (Hai et al., 2014) able to discard opinion feature "amazon", when executed considering "cellphone" as domain data set and "hotel" as non-domain data set during the experiment. However, IEDR (Hai et al., 2014) fails to discard non-domain feature "service", as it equally appears in both "cellphone" as well as "hotel" data sets. On the contrary, KEAE (Luo et al., 2019) utilizes the Probase and Wordnet to narrow the aspect space and therefore substantially removes the non-domain-features.

Among the heuristic approaches, the DOFM (δ_μ) performs consistently well for all data sets. However, it is to note that DOFM (δ_M) also outperforms the state-of-the-art existing approaches in spite of reduced performance of DOFM (δ_M) compared to DOFM

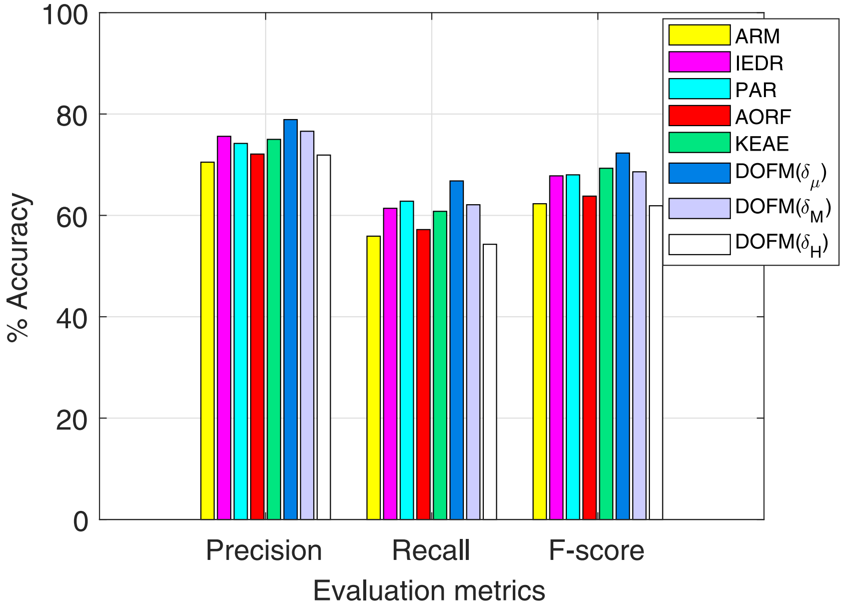


Fig. 14. Results of opinion feature set mining for television data set. The results are statistically significant with *T-Test*, *P-Values* < 0.05.

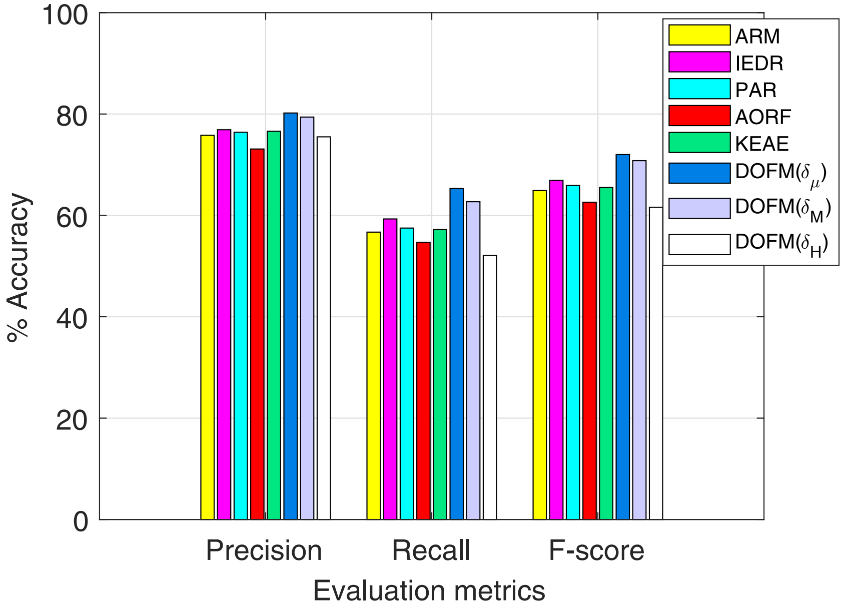
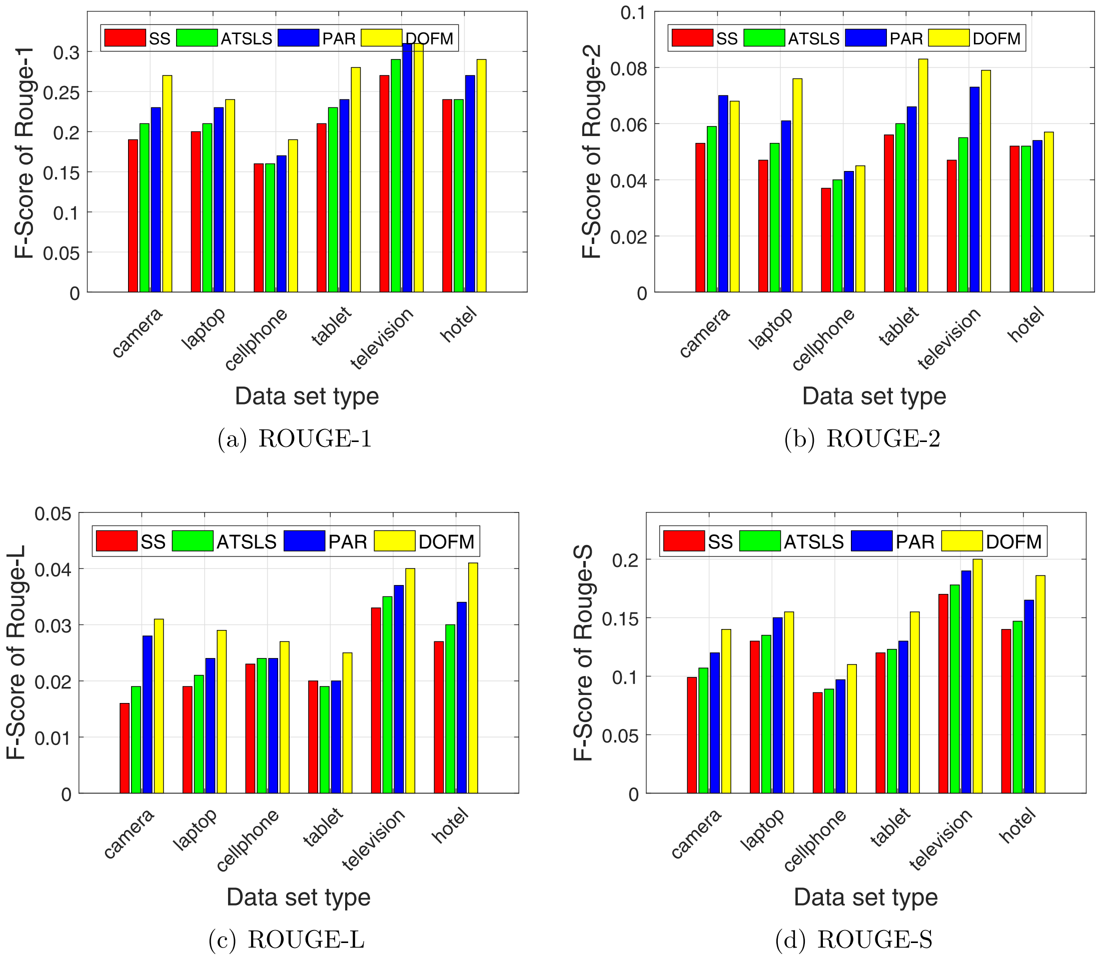


Fig. 15. Results of opinion feature set mining for hotel data set. The results are statistically significant with *T-Test*, *P-Values* < 0.05.

(δ_μ). The performance of DOFM (δ_H) is observed inconsistent, and in many instances its performance is not at par with the existing approaches. For example, in “cellphone” data set, DOFM (δ_H) reports reduced performance under all evaluation parameters compared to IEDR (Hai et al., 2014); whereas in “hotel” data set, the DOFM (δ_H) performs poorly compared to IEDR (Hai et al., 2014) and PAR (Zha et al., 2014) across the parameters.

5.7. Quality evaluation of output extractive summary

In this section, DOFM summaries are evaluated in terms of quality improvement. Domain experts are requested to extract the informative sentences in the decreasing order of the informativeness to formulate summary for each of the six data sets. From the domain-experts’ summary, Top-10 informative sentences are selected and reference summaries are generated. On the



(a) ROUGE-1
(b) ROUGE-2
(c) ROUGE-L
(d) ROUGE-S

Fig. 16. Quality evaluation of output extractive summaries using, (a) ROUGE-1, (b) ROUGE-2, (c) ROUGE-L, (d) ROUGE-S.

contrary, DOFM formulates the summaries as follow. First the Sentence Scoring (SS) approach (Abuobieda, Salim, Albaham, Osman, & Kumar, 2012) is applied and informative sentences are obtained based on sentence position, sentence length, title, and word frequency. Later, sentences without domain features are ignored and the remaining sentences are arranged in decreasing order of their informativeness. Finally, the Top-10 sentences are used to formulate the automatic summary. The DOFM generated summary is evaluated against the state-of-the-art alternatives such as SS (Abuobieda et al., 2012), Automatic Text Summarization using Lexical chain with Semantic-related terms (ATSLS) (Lynn, Choi, & Kim, 2018), and Product Aspect Ranking model (PAR) (Zha et al., 2014). The comparison of the DOFM system with sentence scoring (Abuobieda et al., 2012) is exclusively performed to analyze the quality of extractive summary with and without the presence of domain opinion features.

The evaluation is carried out using ROUGE (i.e., Recall-Oriented Understudy for Gisting Evaluation), a well-known package for automatic evaluation of summaries (Lin, 2004). For the comprehensive evaluation, summaries are evaluated against four different measures such as ROUGE-1 (i.e., uni-gram), ROUGE-2 (i.e., bi-gram), ROUGE-L (i.e., longest common subsequence), and ROUGE-S (i.e., Skip-bigram). The ROUGE-1 and ROUGE-2 is generalized as ROUGE-N (i.e., n-gram) and is defined as shown in Eq. (5).

$$ROUGE-N = \frac{\sum_{S\in\{RSs\}}\sum_{gram_n\in S} Count_{match}(gram_n)}{\sum_{S\in\{RSs\}}\sum_{gram_n\in S} Count(gram_n)} \tag{5}$$

Here, RS_s represents the reference summaries, n represents the length of the n-gram (i.e., $gram_n$), and $Count_{match}(gram_n)$ represents the number of n-grams co-appearing between automatic summary and set of reference summaries.

Fig. 16 shows the performance comparison of extractive summaries generated using DOFM to that of SS (Abuobieda et al., 2012), ATSLS (Lynn et al., 2018), and PAR (Zha et al., 2014) with respect to F-score. Fig. 16a and b shows the F-score of ROUGE-1 and ROUGE-2 for each data set, respectively. Similarly, Fig. 16c and d presents the corresponding performance evaluation for ROUGE-L and ROUGE-S, respectively.

From the results, it can be inferred that the extractive summary generated using DOFM significantly outperforms the one generated using SS (Abuobieda et al., 2012) and ATSLS (Lynn et al., 2018), and consistently shows improvement over the one

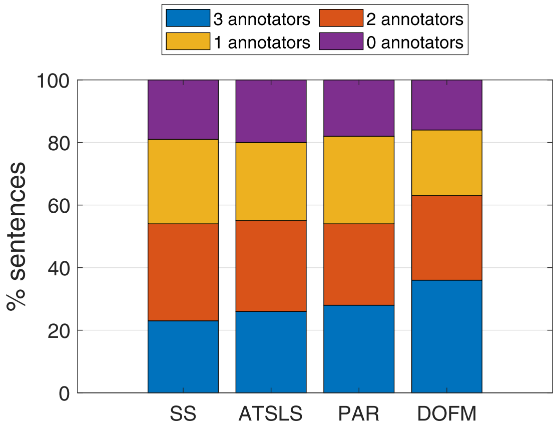


Fig. 17. The error analysis outcome in the form of percentage of sentences selected by expert annotators.

generated using the PAR (Zha et al., 2014). To be specific, DOFM improves the quality of extractive summary over SS (Abuobieda et al., 2012) with average gain of 22.6%, 38.1%, 39.8%, and 26.9% in F-score of ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-S, respectively. Compared to ATSLS (Lynn et al., 2018), the DOFM has the average gain of 18.5%, 27.1%, 34.4%, 22.4% in F-score of ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-S, respectively. Similarly, the DOFM shows the average gain of 10.5%, 11.2%, 15.5%, and 11.0% compared to PAR (Zha et al., 2014) on performance measures ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-S, respectively.

The PAR (Zha et al., 2014) is an aspect (i.e., feature) ranking based model, and therefore it achieves improved performance over SS (Abuobieda et al., 2012). However, in several instances, it fails to correctly discard the non-domain features and wrongly consider them as domain features during the mining process. Contrary to PAR (Zha et al., 2014), ATSLS (Lynn et al., 2018) applies the Näive approach to extracts the aspects (nouns and pronouns as aspects) and therefore it wrongly captures the non-domain aspects as well such as "service" and "amazon" from cellphone data set. This results in the limited performance gain of ATSLS (Lynn et al., 2018) compared to PAR (Zha et al., 2014) and DOFM. The proposed DOFM system effectively prune the non-domain features (aspects) during the clustering process and improves the quality of domain opinion feature set and subsequently improves the quality of extractive summary. The statistical significance of the results is ensured by performing *T-Test* with *P-Values*< 0.05.

5.8. Error analysis

An error analysis is carried out to investigate the quality of the DOFM generated summaries. Three expert annotators were engaged to rank the sentences as a measure of sentence quality. The sentences rejected by all the annotators are considered as errors and those selected by all the annotators are considered as ideal choices.

The error analysis process is repeated for the thirty different summaries the results are averaged. Fig. 17 shows the outcome of the error analysis in the form of percentage of sentences selected by annotators. The results of the proposed DOFM approach is compared with the state-of-the-art alternatives such as Sentence Scoring (SS), Automatic Text Summarization using Lexical chain with Semantic-related terms (ATSLS) (Lynn et al., 2018), and Product Aspect Ranking (PAR). The results show that 36% of sentences in DOFM summaries are selected by all the annotators, which infers that at least 33% of any DOFM summary is highly informative. Moreover, 84% of sentences in DOFM summaries are preferred by at least one of the annotators, which infers the consistent performance of the DOFM. On the contrary, merely 16% of sentences in DOFM summaries are rejected by all the annotators and considered as errors. Among the alternative approaches, SS has the least percentage of ideal sentences followed by ATSLS and PAR. The DOFM generates the extractive summaries by selecting the sentences that contain the domain feature. This puts the DOFM in an advantages position as most end users are interested to know about the various aspects of the product which is conveyed through the domain features. On the contrary, SS suffers from the fact that the selection of the sentence is carried using parameters such as sentence position, sentence length, word frequency, etc. However, the aforementioned parameters do not assure the selection of sentences that contain domain features. Therefore, the generated summaries may not provide informative product-specific aspects and make them less useful to the end users. The ATSLS generated summaries shows the marginal improvement over the SS as ATSLS summaries are based on the product aspects. However, ATSLS follows the naive approach for the aspect extraction such as it considers nouns and pronouns as product aspects, which does not hold in every instances.

6. Conclusion

In this paper, a Domain Feature Miner is designed to mine domain features from the colloquial real-life reviews. Different from the existing approaches, the DOFM engages three empirical observations such as frequency count, grouping semantics, and distributional statistics of features. The extensive experimental evaluation is performed on six publicly available benchmark data sets from the University of Illinois at Urbana–Champaign, which reveals that the proposed DOFM system delivers improved quality of domain features as evident from the improved Precision, Recall, and F-score. Moreover, the quality of extracted domain features is further verified for the application in extractive review summarization using solid performance metric ROUGE. The results of error analysis confirms the noticeable improvement in DOFM generated summaries. The drawback of DOFM is that the parameters such as frequency count, grouping semantics, and distributional characteristics are quantitative in nature and therefore sufficient number of reviews are required to confirm the domain features. In future, we intend to extend the study to identify the domain-specific as well as domain-independent opinion features to benefit the extractive summarization and sentiment polarity in a large size colloquial reviews.

CRediT authorship contribution statement

Hiren Kumar Thakkar: Conceived the idea and developed the algorithms, Analyzed the data, Performed the simulation, Writing - original draft. **Prasan Kumar Sahoo:** Conceived the idea and developed the algorithms, Supervised the work, Revised the manuscript. **Pranab Mohanty:** Designed the simulation outline.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the Ministry of Science and Technology (MOST), Taiwan, under Grant 109-2221-E-182-014 and in part by Chang Gung Medical Foundation, Taiwan under Grant CMRPD 2J0141 and CMRPD 2J0142.

References

Abas, A. R., El-Henawy, I., Mohamed, H., & Abdellatif, A. (2020). Deep learning model for fine-grained aspect-based opinion mining. *IEEE Access*, *8*, 128845–128855.

Abuobieda, A., Salim, N., Albaham, A. T., Osman, A. H., & Kumar, Y. J. (2012). Text summarization features selection method using pseudo genetic-based model. In *Information retrieval & knowledge management (CAMP), 2012 international conference on* (pp. 193–197). IEEE.

Amplayo, R. K., & Song, M. (2017). An adaptable fine-grained sentiment analysis for summarization of multiple short online reviews. *Data & Knowledge Engineering*, *110*, 54–67.

Arabie, P., & Hubert, L. J. (1990). The bond energy algorithm revisited. *IEEE Transactions on Systems, Man, and Cybernetics*, *20*(1), 268–274.

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

Da'u, A., Salim, N., Rabiu, I., & Osman, A. (2020). Weighted aspect-based opinion mining using deep learning for recommender system. *Expert Systems with Applications*, *140*, Article 112871.

Dragoni, M., Federici, M., & Rexha, A. (2019). An unsupervised aspect extraction strategy for monitoring real-time reviews stream. *Information Processing & Management*, *56*(3), 1103–1118.

García-Sánchez, F., Colomo-Palacios, R., & Valencia-García, R. (2020). A social-semantic recommender system for advertisements. *Information Processing & Management*, *57*(2), Article 102153.

Hai, Z., Chang, K., Kim, J.-J., & Yang, C. C. (2014). Identifying features in opinion mining via intrinsic and extrinsic domain relevance. *IEEE Transactions on Knowledge and Data Engineering*, *26*(3), 623–634.

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 168–177). ACM.

Jin, W., Ho, H. H., & Srihari, R. K. (2009). OpinionMiner: a novel machine learning system for web opinion mining and extraction. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1195–1204).

Kang, M., Ahn, J., & Lee, K. (2018). Opinion mining using ensemble text hidden Markov models for text classification. *Expert Systems with Applications*, *94*, 218–227.

Kumar, A., & Abirami, S. (2018). Aspect-based opinion ranking framework for product reviews using a spearman's rank correlation coefficient method. *Information Sciences*, *460*, 23–41.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop* (pp. 74–81). Barcelona, Spain: Association for Computational Linguistics.

Luo, Z., Huang, S., & Zhu, K. Q. (2019). Knowledge empowered prominent aspect extraction from product reviews. *Information Processing & Management*, *56*(3), 408–423.

Lynn, H. M., Choi, C., & Kim, P. (2018). An improved method of automatic text summarization for web contents using lexical chain with semantic-related terms. *Soft Computing*, *22*(12), 4013–4023.

Nasar, Z., Jaffry, S. W., & Malik, M. K. (2019). Textual keyword extraction and summarization: State-of-the-art. *Information Processing & Management*, *56*(6), Article 102088.

Pham, D.-H., & Le, A.-C. (2018). Learning multiple layers of knowledge representation for aspect based sentiment analysis. *Data & Knowledge Engineering*, *114*, 26–39.

Poria, S., Cambria, E., & Gelbukh, A. (2016). Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems*, *108*, 42–49.

Sim, J., & Wright, C. C. (2005). The kappa statistic in reliability studies: use, interpretation, and sample size requirements. *Physical Therapy*, *85*(3), 257–268.

Wang, H., & Hong, M. (2019). Supervised hebb rule based feature selection for text classification. *Information Processing & Management*, *56*(1), 167–191.

Wang, H., Lu, Y., & Zhai, C. (2011). Latent aspect rating analysis without aspect keyword supervision. In *Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 618–626). ACM.

Wang, Y., Sun, A., Huang, M., & Zhu, X. (2019). Aspect-level sentiment analysis using as-capsules. (pp. 2033–2044).

Wu, S., Wu, F., Chang, Y., Wu, C., & Huang, Y. (2019). Automatic construction of target-specific sentiment lexicon. *Expert Systems with Applications*, *116*, 285–298.

Wu, C., Wu, F., Wu, S., Yuan, Z., & Huang, Y. (2018). A hybrid unsupervised method for aspect term and opinion target extraction. *Knowledge-Based Systems*, *148*, 66–73.

Xia, H., Yang, Y., Pan, X., Zhang, Z., & An, W. (2019). Sentiment analysis for online reviews using conditional random fields and support vector machines. *Electronic Commerce Research*, 1–18.

Xu, F., Pan, Z., & Xia, R. (2020). E-commerce product review sentiment classification based on a naïve Bayes continuous learning framework. *Information Processing & Management*, Article 102221.

Zha, Z.-J., Yu, J., Tang, J., Wang, M., & Chua, T.-S. (2014). Product aspect ranking and its applications. *IEEE Transactions on Knowledge and Data Engineering*, *26*(5), 1211–1224.

Zhang, Y., Barzilay, R., & Jaakkola, T. (2017). Aspect-augmented adversarial networks for domain adaptation. *Transactions of the Association for Computational Linguistics*, *5*, 515–528.

Zhao, R., & Mao, K. (2017). Fuzzy bag-of-words model for document representation. *IEEE Transactions on Fuzzy Systems*, *26*(2), 794–804.