ARTICLE IN PRESS

Knowledge-Based Systems xxx (xxxx) xxx

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Dual memory network model for sentiment analysis of review text☆

JiaXing Shen [c,1], Mingyu Derek Ma [b,c,1], Rong Xiang [c], Qin Lu [c], Elvira Perez Vallejos [a], Ge Xu [d], Chu-Ren Huang [e], Yunfei Long [a,c,d,*]

[a] School of Medicine, University of Nottingham, UK
[b] Department of Computer Science, University of Southern California, United States
[c] Department of Computing, Hong Kong Polytechnic University, Hong Kong
[d] Fujian Provincial Key Laboratory of Information Processing and Intelligent Control, Minjiang University, Fuzhou 350108, China
[e] Department of Chinese and Billigual Studies, Hong Kong Polytechnic University, Hong Kong

ARTICLE INFO

Article history:
Received 14 February 2019
Received in revised form 9 August 2019
Accepted 27 August 2019
Available online xxxx

Keywords:
Network embedding
Heterogeneous network
Attention mechanism
Text processing

ABSTRACT

In sentiment analysis of product reviews, both user and product information are proven to be useful. Current works handle user profile and product information in a unified model which may not be able to learn salient features of users and products effectively. In this work, we propose a dual user and product memory network (DUPMN) model to learn user profiles and product information for reviews classification using separate memory networks. Then, the two representations are used jointly for sentiment analysis. The use of separate models aims to capture user profiles and product information more effectively. Comparing with state-of-the-art unified prediction models, evaluations on three benchmark datasets (IMDB, Yelp13, and Yelp14) show that our dual learning model gives performance gain of 0.6%, 1.2%, and 0.9%, respectively. The improvements are also deemed very significant measured by *p-values*.

1. Introduction

Review text is often meant to express the sentiments of individuals towards a product or a service. Recognizing the underlying sentiment expressed in a piece of text is essential to understand the full meaning of the text. The sentiment analysis community is increasingly interested in using natural language processing (NLP) techniques as well as emotion theories to identify sentiments expressed in the review text.

Review text written by a person is likely to be subjective or biased towards his/her own preferences. Review text can be written for commercial products such as cell phones, camera, personal computers etc. This paper focuses on how user profiles can be better incorporated in sentiment analysis for review text. Review comments can influence sentiment analysis results for review text [1,2]. Lenient users tend to give higher ratings than finicky ones even if they review the same products. On the other hand, popular products do receive higher ratings than those unpopular ones because the aggregation of user reviews still shows the difference in opinions for different products [3].

Recent works in emotion analysis have attempted to incorporate user profile information together with product information in neural network models [4–8] including Convolutional Neural Network (CNN) [4], Recurrent Neural Network (RNN) [9], Long Short-Term Memory (LSTM) [6], and Memory Network [7]. Among these models, the memory network model [7] regarded as the state-of-the-art method, this model allows much larger context by using an array of individually learned document representation with the view to capture information at a much larger context. In the proposed memory network model by [7], user profiles and product information are incorporated together in a single memory. However, all previous works handle user profile and product information in a unified model non-discriminantly. User profiles and product information are not independent of each other in opinion analysis. A user profile is encoded in all the documents he/she writes and opinions of a product also encoded in comments written by all users. Yet, putting such information together in a unified model may not be able to capture user profile or product information appropriately. Even though both user and product information play crucial roles in sentiment analysis, they are fundamentally different as indicated by the following example.

In a review about movie video v posted by user u, u said *"The movie is so good and touching"*. From the perspective of user

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.knosys.2019.105004.

* Corresponding author at: School of Medicine, University of Nottingham, UK.
E-mail addresses: jiaxing.shen@connect.polyu.hk (J. Shen), Yunfei.Long@nottingham.ac.uk (Y. Long).

[1] These two authors contributed equally.

https://doi.org/10.1016/j.knosys.2019.105004
0950-7051/

Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.

profiles, if user (represented as u) is a relatively finicky person, he may only give 2 stars out of 5 even if his review content is quite positive. If u is a lenient person, his rating is likely to be 5 stars. From the perspective of video quality rated by all users, the topic of v may be easy to touch people and make people emotional, even most of the reviews about v are very positive, but maybe the actual quality is only 2 stars out of 5.

Reviews written by a user can be affected by user profiles which are more subjective whereas reviews for a product are useful only if they are from a collection of different reviewers, because we know individual reviews can be biased. The popularity of a product tends to reflect the general impression of a collection of reviewers as an aggregated result. Therefore, sentiment prediction of a product should give dual consideration to individual users as well as all reviews for a product as a collection. However, to process user profile and product information in a unified model may not be able to learn salient features of users and products effectively.

We propose a model to learn user profiles and product information as two separate collections using separate memory networks before making a joint prediction on sentiment classification. Firstly, we represent a collection of user profiles by an array of individual profiles in a memory network model. To capture a larger context of products, we also build a memory of product reviews as an array to include a larger context of products. Once both user profile memory and product memory are learned, they are incorporated together to learn a joint representation for opinion analysis. Our proposed model is referred to as the Dual User and Product Memory Network (DUPMN) model because we have two separately built memory networks: a user memory network (UMN) and a product memory network (PMN), both are based on document representation of user comments and product reviews.

To validate the effectiveness of our proposed model, evaluations are conducted on three bench-marking review datasets from IMDB and Yelp data challenge (including Yelp 13 and Yelp 14) [9]. Experimental results show that our algorithm outperforms baseline methods by large margins. Compared to the state-of-the-art method, DUPMN makes 0.6%, 1.2%, and 0.9% increase in accuracy with *p-values* 0.007, 0.004, and 0.001 in the three benchmark datasets respectively. This shows that leveraging user profile and product information separately can indeed extract salient features of user profiles and product information more effectively for sentiment predictions.

The rest of this paper is organized as follows. Section 2 provides a scoping review on recent progress in sentiment analysis and memory network. Section 3 introduces our proposed DUPMN model in details. Section 4 presents evaluation results on three commonly used review datasets compared to state-of-the-art methods. Section 5 concludes this paper and gives some future directions to give more consideration of individual bias in opinion analysis.

2. Related works

Related works are grouped under two sections. The first section introduces neural network models used in sentiment analysis. The second section introduces works in sentiment analysis that make use of user/product information.

2.1. Neural network models

In recent years, the use of deep learning based models has greatly improved the performance of sentiment analysis. Commonly used models include Convolutional Neural Network (CNN) [10], Recursive Neural Network (ReNN) [11], and Recurrent Neural Network (RNN) [12]. RNN naturally benefits sentiment classification because of its ability to capture sequential information in a text. However, standard RNN suffer from gradient vanishing or exploding problem [13] where gradients may grow or decay exponentially over long sequences. Long Short-Term Memory (LSTM) models are able to handle the problem of gradient vanishing. An LSTM model provides a gated mechanism to keep the long-term memory. Each LSTM layer is generally followed by mean pooling and the output is fed into the next layer. Experiments in datasets which contain sentences and long documents demonstrate that the LSTM model outperforms the traditional RNN [9,14]. [15] also proposed a model to utilize neural network models for multi-domain sentiment analysis.

Attention mechanism, an NLP technique, is also added to LSTM models to highlight more important text segments at both sentence-level and document-level. Attention models can be built from text in local context [16], user/product information [6, 17,18], and other information such as cognition grounded eye tracking data [19]. LSTM models with attention mechanism are currently the state-of-the-art models in document-level sentiment analysis tasks [6,19]. Memory network is a type of neural networks designed to handle larger context for a collection of documents. Memory networks introduce inference components combined with a socalled long-term memory component [20]. The long-term memory component is a large external memory to represent data as a collection. This collective information can contain either local context [21] or external knowledge [22]. It can also be used to represent the context of users and products globally [23]. Dou [7] uses a memory network model in document-level sentiment analysis and this work produces a comparable result to the state-of-the-art model [6].

2.2. Incorporating user and product information

Both user profile and product information have crucial effects on sentiment polarities. [24] introduces a method to extract user profile from review information. [5] proposes a model by incorporating user and product information into a CNN network for document-level sentiment classification. User ids and product names are included as features in a unified document vector using the vector space model such that document vectors capture important global clues including individual preferences and product information. However, this method considers word-level preference only and an algebraic based representation model has very limited power to capture generalized highlevel semantic information [6].

Gui et al. [4] introduces an inter-subjectivity network to link users to terms they used as well as the polarities of the terms. The network aims to learn writer embeddings which are subsequently incorporated into a CNN network for sentiment analysis. [6] proposes a model to incorporate user and product information into an LSTM model which has an attention mechanism. This model is reported to produce state-of-the-art result in three benchmark datasets (IMDB, Yelp 13, and Yelp 14). Existing works demonstrate the effectiveness of utilizing user profile and product information in sentiment analysis task, but most of them consider user profile and product information as a united group of features.

3. User and product memory network model

In this section, we explain our proposed Dual User and Product Memory Network (DUPMN) model. In DUPMN, document representation is first learned by a hierarchical LSTM network to obtain comprehensive both at the sentence level and at the document level [25]. Then, a dual memory network method is used to train user profiles and product reviews separately using

Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.

the same memory network model. Both of user and product memory networks are joined together to predict sentiments for documents. The code of our model is publicly available at https://github.com/derekmma/DUPMN.

3.1. Task definition

Let D be the set of review documents for sentiment classification. Let U be the set of users, and P be the set of products. For each review document d ($d \in D$), user u ($u \in U$) is the writer of d on product p ($p \in P$). Let $U_u(d)$ be all documents posted by u and $P_p(d)$ be all reviews of p. Thus, $U_u(d)$ and $P_p(d)$ respectively define user context and product context of d. For simplicity, we use $U(d)$ and $P(d)$ directly. The goal of a sentiment analysis task is to predict the sentiment label for each d.

3.2. Document embedding

Document embedding is a new representation form for words using a dense vector. Since review documents for emotion classification such as restaurant reviews or movie comments are normally very long, an appropriate method is needed to embed the documents properly to speed up the training process and achieve better representation. Inspired by the work of [6], a hierarchical LSTM network is used in DUPMN to obtain embedding representations of documents. The hierarchical LSTM has two layers. We use pre-trained word embeddings as the input to the first layer. The first layer is used to obtain sentence representation by the hidden state of an LSTM network. The second layer is used to obtain document-level representation with sentence-level representation as input. User and product attentions are included in the network so that all salient features are included in document representation. For a document d, its embedding is denoted as $\vec{d}$. $\vec{d}$ is a vector representation with dimension size n. In this paper, n takes the common length of 300 used for most embedding learning. In principle, the embedding representation of user context of d, denoted by $\hat{U}(d)$, and product context $\hat{P}(d)$ vary depending on d. The $\hat{U}(d)$ and $\hat{P}(d)$ are dynamic during the training and testing process for each input document d. For easy matrix calculation, we take memory size m defined as number of document vectors in external memories as our model parameter so that $\hat{U}(d)$ and $\hat{P}(d)$ are two fixed $n \times m$ matrices.

3.3. Memory network structure

Inspired by the successful use of memory networks in language modeling, question answering, and emotion analysis [26, 9,7], we propose our DUPMN by extending a single memory network model to two memory networks to capture the different features from two different perspectives. In this way, we analyze different influences from users' perspective and products' perspective separately. Most of sentiment analysis task do not provide additional user profile or product information as additional data fields since they are hard to obtain due to the privacy concern. Therefore, a way to learn corresponding user or product features without requiring additional properties is needed. By utilizing memory network structure, the model would be able to construct an abstract feature environment for user profile and product information respectively, so that the network can learn information from both user and product side with just the pure review text. Since user and product characteristics are reflected in the related review posts by this user and for this product, we use $\hat{U}(d)$ and $\hat{P}(d)$ mentioned before to simulate the feature environment.

The structure of the memory network model is shown in Fig. 1. The use of 3 hops in Fig. 1 is indicative only. Generally speaking, a memory network can have K computational hops and K should be an experimentally selected parameter.

The DUPMN model has two separate memory networks: a User Memory Network (UMN) and a Product Memory Network (PMN). In our model, each hop in a memory network includes an attention layer $Attention_i$ and a linear addition Σ_k. Since the external memory $\hat{U}(d)$ and $\hat{P}(d)$ have the same structure, we use a generic notation $\hat{M}$ to denote them in the following explanations. Each document vector $\vec{d}$ is fed into the first hop of the two networks ($\vec{d}_0 = \vec{d}$). Each $\vec{d}_{k-1}(\mathrm{k} = 1 \ldots \mathrm{K}-1)$ passes through the current attention layer using an attention mechanism defined by a softmax function to obtain the attention weights $\vec{p}_k$ for document d:

$$\vec{p}_k = Softmax(\vec{d}_{k-1}^{T} * \hat{M}). \tag{1}$$

The vector of attention weights $\vec{p}_k$ reflects similarity between each document in external memory and the input document. The similarity will be higher when two documents are semantical closer which can be illustrated by more similar document-level representations. Then attention weighted vector $\vec{a}_k$ is obtained by

$$\vec{a}_k = \sum_{i=0}^{m} p_{ki} * \vec{M}_i. \tag{2}$$

The documents in external memory with higher p_{ki} is going to take more weights when producing the attention weighted vector $\vec{a}_k$. In other words, related documents in external knowledge that similar to the input document d have more attention in attention layers. $\vec{a}_k$ is then linearly added to $\vec{d}_{k-1}$ to produce the output of this hop as $\vec{d}_k$.

After completing the Kth hop (last hop), the output $\vec{d}_K^u$ in UMN and $\vec{d}_K^p$ in PMN are joined together using a weighted mechanism to produce the output of DUPMN, $Output_{DUPMN}$, as given below:

$$Output_{DUPMN} = w_U \vec{W}_U \vec{d}_K^u + w_P \vec{W}_P \vec{d}_K^p. \tag{3}$$

Two different weight vectors $\vec{W}_u$ and $\vec{W}_p$ in Formula (3) can be trained for UMN and PMN. w_U and w_P are two constant weights to reflect the relative importance of user profile $\vec{d}_K^u$ and product information $\vec{d}_K^p$ and they are also learned through the training process.

The parameters in the model include $\vec{W}_U$, $\vec{W}_P$, w_U and w_P. By minimizing the loss, those parameters can be optimized. The final emotion label prediction is obtained through a *Softmax* layer. The loss function is defined by the cross entropy between the prediction from $Output_{DUPMN}$ and the ground truth labels.

In summary, when a document input to the network, its representation is first obtained as input to the memory network. In the meantime, representations of related user and product documents retrieved according to user ID and product ID will be stacked together as user and product external knowledge. Then the final prediction review score result is obtained. The components we used in the DUPMN model are selected to meet the needs for sentiment classification task. Firstly, to learn semantic features from the review text, we use advanced hierarchical LSTM network which already shown great performance in previous works. Then, to tackle the challenge to extract user and product features without additional related data, memory network enables us to extract features using only pure review text. Finally, the proposed dual memory network design empowers the model to learn features from user profile and product information in separate environment and extract more comprehensive features.

4. Experiment and result analysis

Performance evaluations are conducted on three datasets. The performance of DUPMN is compared against a set of commonly used baseline methods including the state-of-the-art LSTM based methods.

Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.

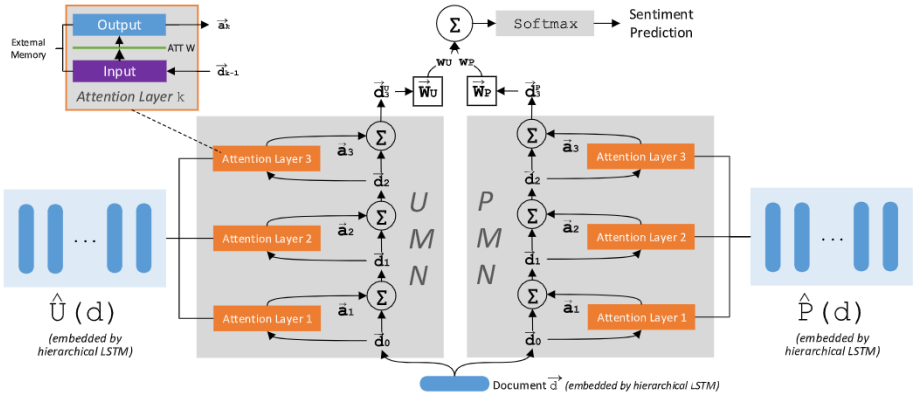


Fig. 1. Structure for proposed DUPMN model.

Table 1
Statistics of the three benchmark datasets.

	IMDB	Yelp13	Yelp14
#class	10	5	5
#doc	84,919	78,966	231,163
#users	1310	1631	4818
#products	1635	1631	4194
Av sen. len	24.56	17.37	17.25
Av docs/user	64.82	48.41	47.97
Av docs/prod	51.93	48.41	55.12
#p(0–50)	1223	1299	3150
#p(50–100)	318	254	749
#p(100–150)	72	56	175
#p(150–200)	22	24	120

Table 2
Parameters in the model.

	HLSTM	Memory network
Batch size	16	32
Initial learning rate	0.01	0.001
Dimension of hidden states	200	NA
Embedding size for sentence	200	200
Embedding size for document	200	200
Optimizer	Adam	Adam
Default memory size	NA	100

4.1. Datasets and evaluation matrix

The three benchmarking datasets include movie reviews from IMDB, restaurant reviews from Yelp13 and Yelp14 developed by [5]. These three datasets represent typical text for sentiment analysis and they are suitable for our task. In these datasets, each record contains three parts: the pure review text, ID of the user who posted this review and the ID of the product which this review is about. Since the reviews are about a restaurant or movie, so the content is concentrated around a product which makes product-related features useful. Provided user and product IDs enable us to retrieve related documents about the user and product to use as memory documents. These characters make them suitable datasets for us to evaluate how user and product information affect the sentiment analysis performance. What is more, they are widely used by previous works on sentiment analysis [27,11,28,6,19,14,7,4] so that we can better compare the performance of our model with existing ones.

All datasets are tokenized using the Stanford NLP tool [29]. Table 1 lists some useful statistics of the datasets including the number of class labels, the number of documents, the average length of sentences, the average number of documents per user, and the average number of documents per product. The last four rows in Table 1 show the distribution of the total number of posts for different products. For example, #p(0–50) means the number of products which have reviews from size 0 to 50. According to [30], postings in social networks by both users and products follow the long tail distribution [30]. Fig. 2 shows that in our three datasets, distribution of data indeed follows the long-tail distribution. The total number of reviews mostly falls within 1–100 per user or product. We split train/development/test sets at the rate of 8:1:1, following the same setting in [31] and [6]. The best configuration by the development datasets is used for the test set to obtain the final result.

Performance metrics used in our evaluations include Accuracy, MAE and RMSE. Let T be the number of correct predictions; N be the size of the testing set, and py_i and gy_i are the prediction result and ground truth for each training and testing record, respectively. Then, the three performance measures are defined as follows:

$$Accuracy = \frac{T}{N} \tag{4}$$

$$MAE = \frac{\sum_i |py_i - gy_i|}{N} \tag{5}$$

$$RMSE = \sqrt{\frac{\sum_i (py_i - gy_i)^2}{N}} \tag{6}$$

4.2. Parameter settings

Different sets of initial parameters are tested to seek for the optimized network settings for both document embedding hierarchical LSTM network and memory networks. The parameters used in our proposed model for both the hierarchical LSTM model and memory network model are shown in Table 2.

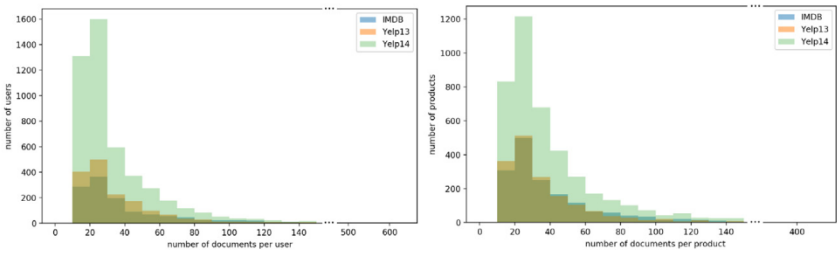
For each dataset, we pre-trained word embeddings using Skip-Gram [32] first, and those embedding will be constantly updated through training.

4.3. Baseline methods

In order to make a systematic comparison, three groups of baselines are used in the evaluation. The first group includes simple baseline methods using commonly used linguistic features. Below is the list of Group 1 methods:

- **Majority**: A simple majority classifier based on sentence labels.
- **Trigram**: An SVM classifier using unigram/bigram/trigram as features.

Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.



(a) Statistic of documents # per user (b) Statistic of documents # per product

Fig. 2. Number of documents per user/product for three datasets.

- **Text feature**: An SVM classifier using word level and context level features, such as n-gram and sentiment lexicons.
- **AvgWordvec**: An SVM classifier that takes the average of word embeddings in Word2Vec as document embedding.

All feature sets in the first group of methods except Majority use the SVM classifier.

The second group includes recent sentiment classification algorithms which are top performers for review text including those state-of-the-art models without using user/product information. Below gives the list of Group 2 methods:

- **SSWE** [27] — An SVM model using sentiment specific word embedding.
- **RNTN+RNN** [11] — A Recursive Neural Tensor Network (RNTN) to represent sentences trained using RNN.
- **CLSTM** [28] — A Cached LSTM model to capture overall semantic information in long text.
- **LSTM + LA** [6] — A state-of-the-art LSTM using local context as attention mechanism in both sentence level and document level.
- **LSTM+CBA** [19]— A novel using cognition based data to build attention mechanism.

The third group includes recent state-of-the-art models using both user and product information. Below is the list of Group 3 methods:

- **UPNN** [14] — A CNN based method to include both user and product information for sentiment classification at document level. Three conversational filters with width 1, 2 and 3 are used to encode the semantics of unigrams, bigrams and trigrams and produce sentence representation finally. Learning rate is set to 0.03, 200-dimensional sentiment-specific word embeddings for each dataset are learned separately.
- **UPDMN** [7] — A memory network based method for document-level sentiment classification by including user and product information by a unified model. Documents are embedded into vectors using a LSTM model by averaging the hidden state of each word. Hop 4 gives best results for IMDB and Yelp13 datasets, and hop 5 gives the best performance for Yelp14 dataset. We use best hop configuration for each datasets.
- **InterSub** [4] — A CNN based method which makes use of network embedding of user and product information. Word embeddings are randomized with 200 dimension, user embedding dimension and product embedding dimension are both 200.
- **LSTM+UPA** [6] — the state-of-the-art LSTM based method which includes both local context based attentions and user/product in the attention mechanism at both sentence level and document level. Word embeddings are pre-trained with 200 dimension, user embedding dimension and product embedding dimension are both 200.

Table 3
Experimental results of DUPMN and comparison models.[3]

	IMDB			Yelp13			Yelp14		
Model	Acc	RMSE	MAE	Acc	RMSE	MAE	Acc	RMSE	MAE
Majority	0.196	2.495	1.838	0.392	1.097	0.779	0.411	1.06	0.744
Trigram	0.399	1.783	1.147	0.577	0.804	0.487	0.569	0.814	0.513
TextFeature	0.402	1.793	1.134	0.572	0.800	0.490	0.556	0.845	0.520
AvgWordvec	0.304	1.985	1.361	0.530	0.893	0.562	0.526	0.898	0.568
SSWE	0.312	1.973	N/A	0.549	0.849	N/A	0.557	0.851	N/A
RNTN+RNN	0.400	1.734	N/A	0.574	0.804	N/A	0.582	0.821	N/A
CLSTM	0.421	1.549	N/A	0.592	0.729	N/A	0.637	0.686	N/A
LSTM + LA	0.443	1.465	N/A	0.627	0.701	N/A	0.637	0.686	N/A
LSTM + CBA	0.489	1.365	N/A	0.638	0.697	N/A	0.641	0.678	N/A
UPNN(K)	0.435	1.602	0.979	0.608	0.764	0.447	0.596	0.784	0.464
UPDMN(K)	0.465	1.351	0.853	0.613	0.720	0.425	0.639	0.662	0.369
InterSub	0.476	1.392	N/A	0.623	0.714	N/A	0.635	0.690	N/A
LSTM+UPA	0.533	1.281	N/A	0.650	0.692	N/A	0.667	0.654	N/A
DUPMN	**0.539**	**1.279**	**0.734**	**0.662**	**0.667**	**0.375**	**0.676**	**0.639**	**0.351**

Our proposed model is labeled as **DUPMN**. In addition, there are two variations to examine the effect of user profiles and project information separately. The first variation only includes user profiles in the memory network, denoted as **DUPMN-U**. The second variation only uses product information, denoted as **DUPMN-P**.

4.4. Experimental results and discussion

Five sets of experiments are conducted. The first experiment compares DUPMN with other sentiment analysis methods. The second experiment evaluates the effectiveness of different hop size K in our memory networks. The third experiment evaluates the effectiveness of UMN and PMN in different datasets. The fourth experiment examines the effect of memory size m on the performance of DUPMN. The fifth experiment examines the effect of using different document representation models. For baseline methods in Group 2 and Group 3, their reported results are used. We also provide the p-value of our model against the state-of-the-art model **LSTM+UPA** [6] by comparing the result of 10 random tests[2] in t-test.[3]

4.4.1. Compare with other sentiment analysis methods

Table 3 shows the result of the first experiment. DUPMN uses one hop configuration (the best performer) with m being set at 100, a commonly used memory size for memory networks. Generally speaking, Group 2 performs better than Group 1. This

2 We re-run experiment based on their public available code on GitHub (https://github.com/thunlp/NSC).

3 http://www.statisticshowto.com/probability-and-statistics/t-test/

3 Overall best results are marked in bold; best results for each group are underlined in the table.

Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.

Table 4
Evaluation of different memory network hops and user and product information utilization.[4]

	IMDB			Yelp13			Yelp14		
	Acc	RMSE	MAE	Acc	RMSE	MAE	Acc	RMSE	MAE
DUPMN-U(1)	0.536	1.273	0.737	0.656	0.687	0.380	0.667	0.655	0.361
DUPMN-U(2)	0.526	1.285	0.748	0.653	0.689	0.382	0.665	0.661	0.369
DUPMN-U(3)	0.524	1.295	0.754	0.651	0.692	0.388	0.661	0.667	0.374
DUPMN-P(1)	0.523	1.346	0.769	0.660	0.668	0.370	0.670	0.649	0.357
DUPMN-P(2)	0.517	1.348	0.775	0.656	0.680	0.380	0.667	0.656	0.364
DUPMN-P(3)	0.512	1.356	0.661	0.651	0.699	0.388	0.661	0.661	0.370
DUPMN(1)	**0.539**	**1.279**	**0.734**	**0.662**	**0.667**	**0.375**	**0.676**	**0.639**	**0.351**
DUPMN(2)	0.522	1.299	0.758	0.650	0.700	0.390	0.667	0.650	0.359
DUPMN(3)	0.502	1.431	0.830	0.653	0.686	0.382	0.658	0.668	0.371

is because Group 1 uses a traditional SVM with feature engineering [33] and Group 2 uses more advanced deep learning methods proven to be effective in recent studies [34,6]. However, some feature engineering methods are no worse than some deep learning methods. For example, the TextFeature model outperforms SSWE by a significant margin.

When comparing Group 2 and Group 3 methods, it is obvious that user profiles and product information can improve performance as most of the methods in Group 3 perform better than methods in Group 2. This is more obvious in the movie review IMDB dataset which naturally contains more subjectivity. In the IMDB dataset, almost all models with user and product information outperform the text-only models in Group 2 except LSTM+CBA [19]. The two LSTM models that include local attention mechanism in Group 2 do show that attention base methods can outperform methods using user profile and product information. In fact, the LSTM + CBA model using attention mechanism based on cognition grounded eye tracking data in Group 2 outperforms quite a number of methods in Group 3. LSTM + CBA in Group 2 is only inferior to LSTM+UPA in Group 3 because of the additional user profile and production information used in LSTM+UPA.

Most important of all, the DUPMN model significantly outperforms all the baseline methods including the state-of-the-art LSTM+UPA model. By using user profiles and product information in memory networks, DUPMN outperforms LSTM+UPA in all three datasets. In the IMDB dataset, DUPMN makes 0.6% improvement over LSTM+UPA in accuracy with *p-value* of 0.007. DUPMN also achieves lower RMSE value. In the Yelp review dataset, the improvement is even more significant. DUPMN achieves 1.2% improvement in accuracy in Yelp13 with *p-value* of 0.004 and 0.9% in Yelp14 with p−*value* of 0.001, the lower RMSE obtained by DUPMN also indicating that the proposed model can predict review ratings more accurately. DUPMN utilizes past documents of users and products and uses the memory mechanism to learn features from those documents, which enables it to learn past behavior and opinion data in a more comprehensive and direct way.

4.4.2. Effect of memory hops

The second set of experiments evaluates the effectiveness of DUPMN using different numbers of hops K. Table 4 shows the evaluation results. The number in the brackets after each model name indicates the number of hops used.

Comparing performances of variations of DUPMN under different numbers of hops, we find that more hops do not bring benefit. In all the three models, the single hop model obtains the best performance. Unlike video and image information, written text is grammatically structured and contains abstract information such that multiple hops may introduce more information distortion. Another reason may be due to over-fitting by the additional hops.

[4] Best results are marked in bold; second best are underlined in the table.

Table 5
Average combine weight for the three different datasets.

IMDB		Yelp13		Yelp14	
w_U	w_P	w_U	w_P	w_U	w_P
0.534	0.466	0.475	**0.525**	0.436	**0.564**

Table 6
Evaluation of different memory size.

Memory size	IMDB			Yelp13			Yelp14		
	Acc	RMSE	MAE	Acc	RMSE	MAE	Acc	RMSE	MAE
10	0.507	1.550	0.866	0.631	0.729	0.416	0.649	0.684	0.384
20	0.516	1.378	0.824	0.637	0.718	0.403	0.654	0.672	0.377
30	0.520	1.372	0.791	0.643	0.707	0.397	0.658	0.661	0.362
40	0.524	1.367	0.778	0.647	0.695	0.390	0.667	0.658	0.357
50	0.531	1.332	0.769	0.656	0.690	0.384	0.675	0.653	0.353
75	0.535	1.301	0.748	0.660	0.672	0.379	0.674	0.653	0.354
100	0.539	1.279	0.734	0.662	0.667	0.375	0.676	0.639	0.351

4.4.3. Effect of UMN and PMN

Comparing the performance of DUPMN-U and DUPMN-P against DUPMN in Table 4, it shows that the effects of the use of user profile or product information on performance are different. The combined use provides better performance indicating that both of them are useful. Yet their difference in contribution to the best performance shows that there is a difference in performance contribution. Another observation is that their usefulness is different in different datasets. For the movie review IMDB dataset, which is more subjective, results show that user profile information used in DUPMN-U are more effective as there is a 1.3% gain compared to that of DUPMN-P. However, on restaurant reviews in Yelp datasets, DUPMN-P performs better than DUPMN-U indicating product information is more valuable.

To further examine the effects of UMN and PMN to sentiment classification, we examine the difference of the optimized values of the constant weights w_U and w_P between DUPMN-U and DUPMN-P given in Formula (3). The difference in their values indicates the relative importance of the features learned from the two memory networks. The optimized weights given in Table 5 on the three datasets show that user profiles have higher weights than product information in IMDB dataset because movie review is more related to personal preferences whereas product information has higher weights in the two restaurant review datasets. This result is consistent with the evaluation in Table 4 on DUPMN-U and DUPMN-P.

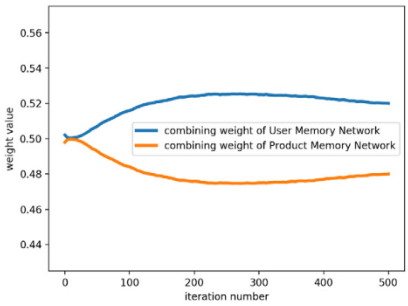
Fig. 3 shows the change of w_U and w_P in the learning process of DUPMN for the IMDB, Yelp13 and Yelp14 datasets. Table 5 shows the average combining weight w_U and w_P for all three benchmark datasets.

The figures of three datasets show two different trends. Fig. 3(a) shows in movie reviews, the weight of user goes up while the weight of product goes down, and the optimized weight shows user profile has higher weight than product information. Figs. 3(b) and 3(c) show a different trend, while the product information has higher weight.

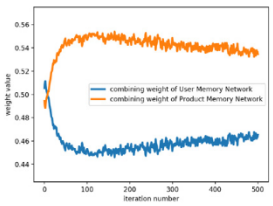
4.4.4. Effect of memory size

According to social network studies, most social network data follows a long tail distribution [35]. If the dimension size to represent social media data is too small, some context information would be lost. On the other hand, too large a dimension size would require more resources in computation and storage without gaining much benefit. Thus, this experiment evaluates the effect of dimension size m in the DUPMN memory networks. Results are given in Fig. 4 and Table 6.

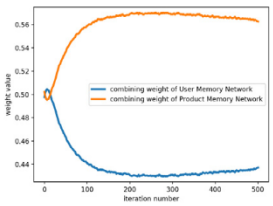
Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.



(a) for IMDB dataset



(b) for Yelp13 dataset



(c) for Yelp14 dataset

Fig. 3. The change of w_U and w_P in the learning process of DUPMN for the three datasets.

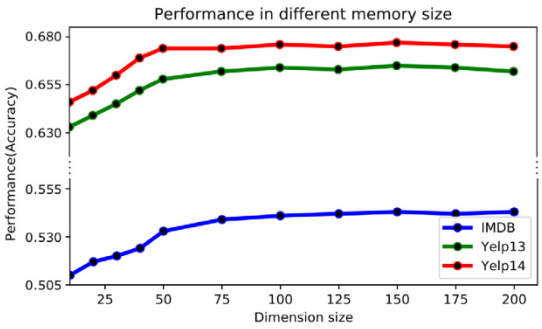


Fig. 4. Effect of different memory sizes for the three datasets.

Fig. 4 shows the result for 1 hop configuration with memory size starting at 10 with 10 points in each increment until size of 50 and in 25 point increment from 50 to 200 to cover most postings. Results show that when memory size increases from 10 to 100, the performance of DUPMN steadily increases. Once it goes beyond 100, DUPMN is no longer sensitive to memory size. This is easy to explain. Note that in Table 1, the average document size is around 50 or so. With long tail distribution, not many documents are available to be included once m reaches 75. With the current computing power and memory capacity, any value between 100–200 would be a reasonable value for m.

4.4.5. Effect of document representation

In our proposed DUPMN model, we need to first select a document representation model. To fully appreciate the performance gain of our model, this experiment examines the use of different document representation models in DUPMN so as to know if the gain in performance is due to the choice of representation method or the dual memory models with appropriate attention mechanism. In this set of experiments, the default method LSTM+UPA used for getting the document representation is replaced by other embedding methods listed in Group 2 and Group 3 of Section 4.3.

Table 7 lists the performance of different document representation models with or without the use of DUPMN. The Accuracy

Fig. 5. Word clouds of reviews for 10 users who give average highest or lowest ratings (above), and 10 products which have average highest or lowest ratings (below) in IMDB.

Fig. 6. Word clouds of reviews for 10 users who give average highest or lowest ratings (above), and 10 products which have average highest or lowest ratings (below) in Yelp13.

improvement by using DUPMN added on the first step models of IMDB dataset range from 0.6% to 3.0%. Improvements in Yelp 13 and Yelp 14 also shows similar results. The improvement indicates that the DUPMN model can incorporate user and product information more effectively regardless of different first step document representation models.

4.5. Feature analysis

This experiment examines features extracted from users as compared to that of products. Feature analysis is conducted in two parts. The first part shows the difference in features extracted by user memory and product memory. The second part examines the use of adjectives in the two memories.

Fig. 5[5] shows two groups of word cloud graphs for IMDB dataset. The two upper sub-figures in Fig. 5 shows two word cloud graphs that demonstrate the word frequency of reviews of the top 10 users giving highest ratings (i.e. lenient raters) and 10 users who give average lowest ratings (i.e. finicky raters) to movies in IMDB. Note that the high-frequency words include both personal feelings and product description but using different polarities. Personal feelings include words such as *like* (positive), *bad* (negative), etc. and movie description words include: *wonderful* (positive), *not great* (negative), etc. By contrast, words used in reviews for 10 highest or lowest rated movies, as shown in the two sub-figures in the bottom of Fig. 5, are more objective, such

[5] Word cloud tool is from (https://www.wordclouds.com/).

Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.

Table 7
Experimental results of DUPMN under different document representation models.

	IMDB			Yelp13			Yelp14		
Model	Acc	RMSE	MAE	Acc	RMSE	MAE	Acc	RMSE	MAE
SSWE	0.312	1.973	N/A	0.549	0.849	N/A	0.557	0.851	N/A
SSWE w/ DUPMN	0.322	1.873	N/A	0.567	0.826	N/A	0.581	0.821	N/A
RNTN + RNN	0.400	1.734	N/A	0.574	0.804	N/A	0.582	0.821	N/A
RNTN + RNN w/ DUPMN	0.430	1.694	N/A	0.601	0.774	N/A	0.612	0.796	N/A
CLSTM	0.421	1.549	N/A	0.592	0.729	N/A	0.637	0.686	N/A
CLSTM w/ DUPMN	0.445	1.459	N/A	0.611	0.719	N/A	0.642	0.681	N/A
LSTM + LA	0.443	1.465	N/A	0.627	0.701	N/A	0.637	0.686	N/A
LSTM + LA w/ DUPMN	0.475	1.438	N/A	0.645	0.660	N/A	0.652	0.670	N/A
LSTM + CBA	0.489	1.365	N/A	0.638	0.697	N/A	0.641	0.678	N/A
LSTM + CBA w/ DUPMN	0.511	1.293	N/A	0.646	0.657	N/A	0.653	0.668	N/A
UPNN(K)	0.435	1.602	0.979	0.608	0.764	0.447	0.596	0.784	0.464
UPNN(K) w/ DUPMN	0.464	1.372	0.869	0.615	0.724	0.427	0.636	0.674	0.374
UPDMN(K)	0.465	1.351	0.853	0.613	0.720	0.425	0.639	0.662	0.369
UPDMN(K) w/ DUPMN	0.474	1.342	0.843	0.625	0.716	0.419	0.647	0.654	0.360
InterSub	0.476	1.392	N/A	0.623	0.714	N/A	0.635	0.690	N/A
InterSub w/ DUPMN	0.496	1.299	N/A	0.643	0.674	N/A	0.649	0.670	N/A
LSTM + UPA	**0.533**	**1.281**	**N/A**	**0.650**	**0.692**	**N/A**	**0.667**	**0.654**	**N/A**
LSTM + UPA w/ DUPMN	**0.539**	**1.279**	**N/A**	**0.662**	**0.667**	**N/A**	**0.676**	**0.639**	**N/A**

Table 8
Adjective frequency table of users and products with 10 highest and 10 lowest ratings in IMDB.

IMDB user				IMDB product			
Highest		Lowest		Highest		Lowest	
Word	Frequency	Word	Frequency	Word	Frequency	Word	Frequency
Great	413	DisLike/hate	566	Great	531	(Not) great	104
Good	145	Good	236	Best	460	Like	95
Best	143	Bad	228	Like	458	Good	86
Excellent	95	Great	138	Most	390	Best	78
Wonderful	94	Better	125	Good	339	Little	58
Classic	93	Original	110	Wonderful	223	Different	41
Fantastic	85	Big	109	Greatest	185	Delicious	39
Funny	72	Real	109	Classic	164	Amazing	34
Brilliant	63	Old	107	New	156	Nice	29
Dead	60	New	103	Old	150	Better	29
Old	58	Best	89	Little	148	Fresh	29
Real	55	Least	88	Perfect	143	Sweet	28
Dark	54	Few	87	Better	135	Perfect	28
Little	53	Funny	87	Same	122	Wonderful	27
Like	52	Dead	86	Real	121	Beautiful	26
Better	52	Stupid	69	Another	118	Before	26
Original	52	Boring	65	Few	115	Favorite	25
Beautiful	45	Black	63	Silent	113	Small	25
Young	45	long	60	Big	112	First	24
Hilarious	44	Salty	57	Young	99	Most	24

as *old*, *new*, *little*, etc. Those words are mainly about the movies themselves rather than personal feelings.

The two restaurant review datasets show different characters. In the two upper sub-figures in Fig. 6, it is hard to distinguish the best and worst raters. Even the worst raters use positive words like *better, great, fresh,* etc in a high frequency. But the product information, which reflects the popularity of the target restaurant in the lower two sub-figures Fig. 6, shows a huge difference between the highest rating products and the lowest rating products. That can partly explain why product memory works better than user memory in the restaurant review datasets.

The second aspect of feature analysis shows the highest 20 adjectives for 10 users giving the highest ratings (lenient raters) and lowest ratings(finicky raters) as well as 10 highest rated product and 10 lowest rated product. Despite the difference between user profile and product information, we observed the huge gap between lenient user and finicky user. Tables 8 and 9 show that in IMDB and Yelp 13, all the 20 highest adjectives for lenient users are positive words, while the most of top 20 adjectives in finicky user are negative words. From the product perspective, the top 20 adjectives for highest rating products are also all positive, while most frequent adjectives for lowest rating products are negative or positive words co-occur with negation (e.g: not). That indicates user profile and product information can provide information to the sentiment prediction model. In the movie review dataset, the user profile is more effective in identifying sentiment than product information, and the restaurant review shows a different trend.

4.6. Case analysis

In this section, we present two cases to show how user or product features are learned and how do they affect the sentiment prediction. The first case analysis is about a piece of selected review text which is for a sci-fi movie with the golden label 10 (most positive). Along with the review text, user ID and product ID are also provided which can be used to find other related reviews posted by this user and about this movie. Please note that if the text is read in isolation, identifying its sentiment is difficult.

Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.

Table 9
Adjective Frequency table of users and products with 10 highest and 10 lowest ratings in YELP 13.

YELP13 user				YELP13 product			
Highest		Lowest		Highest		Lowest	
Word	Frequency	Word	Frequency	Word	Frequency	Word	Frequency
Good	146	Good	143	Great	104	Worst	128
Great	135	Great	97	Like	95	(ot) great	44
Like	90	More	76	Good	86	Bad	34
Best	77	Better	58	Best	78	Nice	22
Wonderful	56	Fresh	51	Little	58	Little	21
Fresh	44	Before	50	Different	41	Different	21
Delicious	37	Hot	43	Delicious	39	Wrong	20
Little	34	Small	42	Amazing	34	Long	19
Nice	33	Little	41	Nice	29	Friendly	17
Amazing	32	Old	41	Better	29	Full	17
Happy	32	Green	33	Fresh	29	Free	17
Tasty	32	Bad	32	Sweet	28	Old	16
Excellent	31	Real	26	Any	28	Hard	15
First	30	Nice	26	Perfect	28	Clean	14
Favorite	30	New	24	Wonderful	27	Big	14
Brilliant	26	High	22	Beautiful	26	Large	13
Few	25	Large	22	Favorite	25	Busy	13
Friendly	23	horrible	21	New	25	Extra	12
Full	22	Happy	20	Small	25	Expensive	12
Hot	22	Special	19	Few	22	Wrong	12

okay , there are two types of movie lovers : the ones who watch one movie every six months and talk about it for the rest of the year, and the ones who actually watch movies all the time. people who belong to the first category , expect everything from a movie, let us say, they expect to see a ' titanic ' every time they go to the cinema. the rest eventually learn to appreciate the good elements of a film , since they know how rare it is to find ' the perfect movie ' ." this movie sucks " ? well, I beg to differ. i mean, it is definitely better than other sci-fi films like 'armageddon' or even 'the phantom menace' no jar-jar here. The audio and visual effects are simply terrific and travolta's performance is brilliant - funny and interesting . What people expect from sci-fi movies is beyond me . When 'starship troopers' was released , absolutely the best space sci-fi movie of the 90's, everyone said it was a bomb. Fortunately , it starts to gain some recognition over the last years , since the release of the dvd. same here, only worse. Why does not anyone care to mention the breath-taking effects or the captivating atmosphere ? what did they expect, a 6 movie saga to satisfy their hunger for sci-fi? At the time these lines are written, the imdb rating for 'battlefield earth' is below 2.5 , which is unacceptable for a movie with such craftsmanship. 'scary movie', possibly the worst movie of all time - including home made movies, has a 6 ! Maybe we should all be a little more subtle when we criticize movies like this and especially sci-fi movies, since they have become an endangered genre. Have you seen any of the major studios produce sci-fi movies lately? Give this movie the recognition it deserves .

In fact, the LSTM + LA model without consideration of user context gives it the rating of 1 (most negative), perhaps because on the surface, there are many negative words such as *unacceptable*, *criticize* and *sucks* even though the reviewer is praising the movie. Since our user memory can learn that the reviewer is a fan of sci-fi movies, our DUPMN model indeed gives the correct rating of 10.

If we have a detailed look of other 27 reviews posted by this user, we can found 48% of the total 27 reviews are about science-fiction movies out of all categories including comedy, action, drama and so on. The average rating is 9 for science-fiction movies reviewed by this user, while the average rating is only 7.42 for non-science-fiction movies from this user. The fraction of reviewing movie categories and higher reviewing score indicate that this user is a science-fiction movie fan. The related review texts that share similar words like this review text mostly have a high ranking score around 8 to 10 since they all belong to science-fiction movie review category. When this document is input to the DUPMN model, these 27 user-related documents will be external memory in the User Memory Network. Those documents with similar words and semantical meaning will obtain higher attention weights and thus lead to higher influences when calculating attention weighted vector. So the final review score prediction from the DUPMN model will tend to have higher correlation with the semantical similar high-ranking documents.

Then we analyze another review from Yelp dataset about a coffee shop shown below. The golden label for this review is 1 (most negative) which is successfully predicted by DUPMN model. While LSTM + LA model gives it the ranking of 4 out of 5 which is pretty positive.

there is a good reason why this place does not post their prices: it is because if they did, most folks would walk right out. this is the first place ever that makes starbucks seem inexpensive. they have only 2 sizes of drinks, a tiny 8 oz and one that is probably 12 oz. originally i wanted a mocha and that was going to be 5 bucks (before tax), so i switched to a regular coffee which costs almost 4 bucks for the 12 oz size. that was my que to laugh this place off as a joke, especially in a college town, and head over to circle k, where there were 12 choices of coffee waiting, all for under 2 bucks. ASU college kids must have more of daddy's money because nobody was buying $5 cups of coffee back in the good ole days. no outdoor seating either.

In this review, the user does not use emotional words or even many adjectives to describe his/her opinion. Instead, many facts such as price and seating are provided. Therefore, it is hard to infer the sentiment of this review by textual features. If we have a look of 110 other reviews about this coffee shop, we can found many of related reviews contain information about facts like price and seating as well. For example, a 1-score review said: *"stupid. we got a little drink for $ 3.50 ... a little ... i mean like 6oz . yeah , definitely not worth it "*. In other documents containing same facts, emotional words and clear negative adjectives provide strong evidence that the appearance of those facts leads to negative review. When the document is input into DUPMN, other reviews about this coffee shop will be retrieved and serve as documents in external memory in Product Memory Network. Since the ones mentioning same kind of content like price and seating are similar with the input review, they will got higher attention and thus the model can learn those negative features about this product and reflect in final prediction result.

Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.

Table 10
Time for training and testing and average convergence iteration numbers. For time, the numbers are in seconds.

	IMDB	Yelp13	Yelp14
Avg time of loading Word emb and dataset metadata	0.33	0.18	0.31
Avg time of loading dataset representations	289.36	129.79	411.97
Avg time of DUPMN model building	0.97	0.97	0.96
Avg time of one training iteration	42.35	28.32	54.42s
Avg time of one time testing	5.44	3.97	8.69
Average convergence iteration	212.31	269.6	209.75

4.7. Complexity analysis

We assume that the length of a document is $|d|$, the size of LSTM hidden state and word embedding is n, the number of LSTM layer is l. As we showed in 3.3, the parameter of external memories $\hat{U}(d)$ and $\hat{P}(d)$ are two fixed $n \times m$ matrices.

Document Embedding: The time complexity of selecting word embedding of a document is $\mathcal{O}(|d|)$, since it only involved a selection operation on word embedding matrix. According to [36] The time complexity of hierarchical LSTM is $\mathcal{O}\left(|d| \times l \times n^2\right)$, since the number of LSTM layer is usually a constant, the result can be reduced as $\mathcal{T}_{DE} = \mathcal{O}\left(|d| \times n^2\right)$

Memory Network: As we described in our paper, each memory hop includes an attention operation and a linear addition operation. The time complexity of attention operation is $\mathcal{O}(m \times n^2)$. The detailed computation procedure is:

$$\begin{aligned}\mathcal{T} &= m \times n^2(\text{attention score}) + m \times n(\text{weighted sum}) \\ &= \mathcal{O}(m \times n^2) + \mathcal{O}(m \times n) \\ &= \mathcal{O}(m \times n^2)\end{aligned} \tag{7}$$

There are h memory hops in our model, the time complexity is accumulated as $\mathcal{O}(h \times m \times n^2)$. The total time complexity of user and product memory is $\mathcal{T}_{MN} = \mathcal{O}(2 \times h \times m \times n^2) = \mathcal{O}(h \times m \times n^2)$. Since the number of memory hops is a constant and usually much less than the square part $\mathcal{O}(n^2)$. The result can be simplified as $\mathcal{T}_{MN} = \mathcal{O}(m \times n^2)$

Output Component: The output component of our model involves two bi-linear matrix multiplication operation. The time complexity is

$$\mathcal{T}_{OC} = \mathcal{O}(2 \times n^3) = \mathcal{O}(n^3) \tag{8}$$

The total time complexity of our model is

$$\begin{aligned}\mathcal{T}_{DE} + \mathcal{T}_{MN} + \mathcal{T}_{OC} &= \mathcal{O}(|d| \times n^2) + \mathcal{O}(m \times n^2) + \mathcal{O}(n^3) \\ &= \mathcal{O}((|d| + m) \times n^2 + n^3)\end{aligned} \tag{9}$$

The time complexity of the model is shown in Table 10. The running time is the average time of five times experiments on a machine with 12 GB memory and a NVIDIA Tesla K80 GPU.

The loading time of dataset representation is related to the size of dataset, and the training and testing time is also dependent on number of available documents in corresponding datasets. The total training time is all of the loading and model building time, as well as number of iterations times the time for one iteration training.

5. Conclusion

In this paper, we present our proposed deep learning method using dual memory network model to make better use of user profiles and product information into sentiment analysis for review text. We argue that user profile and product information are fundamentally different as user profiles contain more subjectivity whereas product reviews, as a collection, contain more salient features of products at the aggregated level.

Based on this hypothesis, two separate memory networks for user context and product context are built at the document-level through a hierarchical learning model. The inclusion of an attention mechanism can capture semantic information more effectively. Learning results from the dual memory networks serve as input to a unified classification model for optimization. Evaluation on three benchmark review datasets shows that our proposed DUPMN model outperforms the current state-of-the-art systems with significant improvements with the p-value of 0.007, 0.004, and 0.001, respectively. We also show that single hop is the most effective setting in the memory network model. Analysis on the contribution of user profile and product information demonstrates that they do have different performance effects on different datasets. In more subjective datasets such as IMDB, the inclusion of user profile information is more important. On the other hand, for more objective datasets such as Yelp data, collective information of restaurant reviews play a more important role in classification.

The proposed DUPMN model also shows some weaknesses which lead the direction for further investigation. Performance of the proposed memory network structure highly relies on the number of related documents available, so performance may drop when related documents for a particular user or product are not sufficient. A future direction worth investigating is to extract more related documents for particular user or product from similar users or products to reduce the performance drop in the situation of lacking external memory documents.

Future work includes two directions. One direction is to explore the contribution of user profiles and product information in sentiment analysis tasks at the aspect level. Another direction is to explore how the knowledge base can be incorporated to further improve the performance of sentiment classification.

Acknowledgments

The work is partially supported by the research grants from Hong Kong Polytechnic University (PolyU RTVU) and GRF, Hong Kong grant (CERG PolyU 15211/14E, PolyU 152006/16E).

Yunfei Long and Elvira Perez Vallejos acknowledge the financial support of the NIHR Nottingham Biomedical Research Centre and NIHR MindTech Healthcare Technology Co-operative.

References

[1] B. Pang, L. Lee, A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts, in: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, 2004, p. 271.

[2] K. Isbister, N. Schaffer, Game Usability: Advancing the Player Experience, CRC Press, 2015.

[3] Y. Long, M. Ma, Q. Lu, R. Xiang, C.-R. Huang, Dual memory network model for biased product review classification, in: Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2018, pp. 140–148.

[4] L. Gui, R. Xu, Y. He, Q. Lu, Z. Wei, Intersubjectivity and sentiment: From language to knowledge, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI, AAAI Press / International Joint Conferences on Artificial Intelligence, 2016, pp. 2789–2795.

[5] D. Tang, B. Qin, T. Liu, Learning semantic representations of users and products for document level sentiment classification, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Vol. 1, 2015, pp. 1014–1023.

[6] H. Chen, M. Sun, C. Tu, Y. Lin, Z. Liu, Neural sentiment classification with user and product attention, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 1650–1659.

[7] Z.-Y. Dou, Capturing user and product information for document level sentiment analysis with deep memory network, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2017, pp. 521–526.

[8] J. Bi, C. Zhang, An empirical comparison on state-of-the-art multi-class imbalance learning algorithms and a new diversified ensemble learning scheme, Knowl.-Based Syst. 158 (2018) 81–93.

[9] D. Tang, B. Qin, T. Liu, Document modeling with gated recurrent neural network for sentiment classification, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 1422–1432.

[10] R. Socher, J. Pennington, E.H. Huang, A.Y. Ng, C.D. Manning, Semi-supervised recursive autoencoders for predicting sentiment distributions, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2011, pp. 151–161.

[11] R. Socher, A. Perelygin, J.Y. Wu, J. Chuang, C.D. Manning, A.Y. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP, Vol. 1631, Citeseer, 2013, p. 1642.

[12] O. Irsoy, C. Cardie, Opinion mining with deep recurrent neural networks, in: EMNLP, 2014, pp. 720–728.

[13] Y. Bengio, P. Simard, P. Frasconi, Learning long-term dependencies with gradient descent is difficult, IEEE Trans. Neural Netw. 5 (2) (1994) 157–166.

[14] D. Tang, B. Qin, T. Liu, Learning semantic representations of users and products for document level sentiment classification, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Beijing, China, 2015, pp. 1014–1023.

[15] M. Dragoni, G. Petrucci, A neural word embeddings approach for multi-domain sentiment analysis, IEEE Trans. Affect. Comput. 8 (4) (2017) 457–470.

[16] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, E. Hovy, Hierarchical attention networks for document classification, in: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016, pp. 1480—1489.

[17] Y. Long, Q. Lu, R. Xiang, M. Li, C.-R. Huang, Fake news detection through multi-perspective speaker profiles, in: Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers), vol. 2, 2017, pp. 252–256.

[18] Z. Wu, X.-Y. Dai, C. Yin, S. Huang, J. Chen, Improving review representations with user attention and product attention for sentiment classification, in: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI-18, 2018.

[19] Y. Long, L. Qin, R. Xiang, M. Li, C.-R. Huang, A cognition based attention model for sentiment analysis, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 473–482.

[20] J. Weston, S. Chopra, A. Bordes, Memory networks, in: Proceedings of the 6th International Conference on Learning Representations, ACM, 2015, pp. 1–15.

[21] R. Das, M. Zaheer, S. Reddy, A. McCallum, Question answering on knowledge bases and text using universal schema and memory networks, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Vol. 2, Association for Computational Linguistics (ACL), 2017, pp. 358–365.

[22] S. Jain, Question answering over knowledge base using factual memory networks, in: Proceedings of the NAACL Student Research Workshop, 2016, pp. 109–115.

[23] D. Tang, B. Qin, T. Liu, Aspect level sentiment classification with deep memory network, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2016, pp. 214–224.

[24] M. Dragoni, A three-phase approach for exploiting opinion mining in computational advertising, IEEE Intell. Syst. 32 (3) (2017) 21–27.

[25] M. Sundermeyer, R. Schlüter, H. Ney, LSTM neural networks for language modeling, in: Proceedings of the 13th Annual Conference of the International Speech Communication Association, International Speech Communication Association (ISCA), 2012, pp. 194–197.

[26] S. Sukhbaatar, J. Weston, R. Fergus, et al., End-to-end memory networks, in: Proceedings of Advances in Neural Information Processing Systems 28, NIPS 2015, Association for Computational Linguistics (ACL), 2015, pp. 2440–2448.

[27] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, B. Qin, Learning sentiment-specific word embedding for Twitter sentiment classification, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vol. 1, Association for Computational Linguistics (ACL), 2014, pp. 1555–1565.

[28] J. Xu, D. Chen, X. Qiu, X. Huang, Cached long short-term memory neural networks for document-level sentiment classification, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics (ACL), 2016, pp. 1660–1669.

[29] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, D. McClosky, The Stanford CoreNLP natural language processing toolkit, in: Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 2014, pp. 55–60.

[30] S. Kordumova, J. van Gemert, C.G. Snoek, Exploring the long tail of social media tags, in: International Conference on Multimedia Modeling, Springer, 2016, pp. 51–62.

[31] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, Q. Mei, Line: Large-scale information network embedding, in: Proceedings of the 24th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee, 2015, pp. 1067–1077.

[32] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in Neural Information Processing Systems, 2013, pp. 3111–3119.

[33] C.-C. Chang, C.-J. Lin, Libsvm: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (3) (2011) 27–36.

[34] Y. Kim, Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, 2014, pp. 1746–1751.

[35] S. McDonald, The long tail and its implications for media audience measurement, J. Advert. Res. 48 (3) (2008) 313–319.

[36] H. Sak, A. Senior, F. Beaufays, Long short-term memory recurrent neural network architectures for large scale acoustic modeling, in: Fifteenth Annual Conference of the International Speech Communication Association, 2014.

Please cite this article as: J. Shen, M.D. Ma, R. Xiang et al., Dual memory network model for sentiment analysis of review text, Knowledge-Based Systems (2019) 105004, https://doi.org/10.1016/j.knosys.2019.105004.