View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by International Journal on Advanced Science, Engineering and Information Technology

International Journal on Advanced Science Engineering Information Technology
Vol.9 (2019) No. 3
ISSN: 2088-5334

Convolutional-NN and Word Embedding for Making an Effective Product Recommendation Based on Enhanced Contextual Understanding of a Product Review

Hanafi[#], Nanna Suryana[*], Abd Samad Bin Hasan Basari[*]

[#]*Department of Computer Science, Universitas Amikom Yogyakarta, Condongcatur, Depok, Sleman, 55283, Indonesia*
E-mail: hanafi@amikom.ac.id

[*] *Fakulti Teknologi Maklumat dan Komunikasi, UTeM Malaka Hang Tuah Jaya, Durian Tunggal Malaka, Malaka 76100, Malaysia*
E-mail: .n.suryana@utem.edu.my; abd.samad@utem.edu.my

***Abstract* — E-commerce is one of the most popular service applications in the world in the last decade. It has become a revolutionary model from traditional shopping transaction to entire internet commerce. E-commerce needs essential artificial intelligence (AI) to provide the customer with information about a product, called a recommendation machine. Collaborative filtering is a model of a recommendation algorithm that relies on rating as the fundamental calculation to make a recommendation. It has been successfully implemented in e-commerce. Even so, this model has a weakness in sparse product data in which the rating number is very low or sparse. Mostly, only less than 3% of the total user population rate a product, leading to the rise of sparse data. A text sentence document is a part of customers' feedback that can be converted into a product rating. According to a traditional approach, bag of word and lexicon model are ignored in a contextual approach. This experiment, it developed a new model to increase the contextuality of text sentences, leading to a more effective rating prediction. We employed a kind of convolutional neural network to generate item latent factor vectors that could be incorporated with probabilistic matrix factorization to make rating prediction. Our method outperformed several previous works based on a metric evaluation using the Root Mean Squared Error (RMSE). In this experiment, we analyzed MovieLens and IMDB datasets, which contained a movie product review.**

***Keywords* — E-commerce; recommender system; convolutional; text sentence; sparse data; product recommendation.**

I. Introduction

A recommendation machine is a form of computer equipment that is an essential machine to create a successful e-commerce business company. When successfully applied, this system will make quite an impact on the sales target achievement in an online transaction. Thus, many large companies in the world have implemented a recommender system to increase their excellent service, and thus to improve customer satisfaction. This system is a part of important machine to boost marketing target. It has been implementing in so many web portal and mobile solution. An example in Netflix company, one of the largest movie online commerce shows that more than 80 percent movie is selling by recommendation system [1], another example on YouTube shows evidence that more than 60 percent video promotes by recommender system [2]. Also, according to Schafer, Konstan, and Riedl [3], marketer representative that use recommender system growth more than 60 percent compared over the traditional marketing model. This evidence shows that the use of recommendation machines has a significant influence on sales and marketing achievement on e-commerce business.

According to Hanafi, Suryana, and Sammad [4], [5], e-commerce recommendation machine algorithm divided into four basic approaches, that is as follows: 1) Content-based filtering: This is a method to generate recommendations based on a particular product categorical technique. This method considers to applied information retrieval approach to creating product suggestion. 2) Knowledge-based filtering: This approach is developed to generate a specific important recommendation. It has special properties in serving information that hardly ever necessary by individual purposes, for instance, housing, leasing, apartment, automobile. 3) Demographic-based filtering: This approach provides item suggestion based on demographic information. 4) Collaborative Filtering (CF): This mechanism is to produce judgment following to customer attitude in the previous work such as rating, product review, comment, testimony, purchasing, etc. CF is considered the most successful technique implemented in a much larger e-commerce company for it can provide a recommendation with some special characters, for examples: providing product fit, giving relevant information, showing high accuracy, and creating serendipity [6]. Commonly, CF applies ratings as explicit feedback for its basic computation in calculating users' similarity using a rating value to develop item suggestion. The big problem in the collaborative model recommendation is less in a number of rating. It is just a few numbers of users giving a rating for the product. Referring to a big data set, such as Movielens [7], it is shown that less than 3 percent of viewers rated the movie they watched. This actual problem is commonly named sparsity problem, which is in a severe problem status also popular as a cold start problem. Finally, if a cold start occurs, it is impossible to acquire the output product recommendation by recommendation machine.

Collaborative Filtering offers several advantages in accuracy, provides serendipitous product information, and shows product diversity over the other approaches such as Content-based, Demographic-based, and Knowledge-based [8][9]. Even so, CF has a severe shortcoming that causes some lack of rating collection from customers [10]. CF relies on rating as its basic computation to obtain the output of product suggestion for the user. There is some attempt proposed by researchers to reduce the problem in sparsity data caused by the limited number of rating. For example, there is an effort made to create a technique aimed to predict the rating of a particular product. One of the efforts by item auxiliary information is to improve the accuracy of rating prediction. Some extra information has been explored to handle sparse data, for instance, audio features in music recommendations [11], [12], color feature extraction for online fashion shop [13], and document recommendation for online news recommendation [14], [15]. IMBD is an example of portal services that provides a product review about a movie coming from its customers.

This study contributes to the development of a novel approach applied in a convolutional neural network to gain an in-depth understanding of a product review from customers. Also, item latent factors are combined with user information into the probabilistic matrix factorization to increase the accuracy of the resulted rating prediction to be then used in the collaborative filtering recommender system.

The earlier collaborative filtering recommender systems, approximately in the mid-'90s, used the traditional approaches such as cosine similarity, Spearman rank, and adjusted cosine. These methods are directed to calculate the similarity among user behaviors to predict the product rating. The traditional approach is straightforward to implement and does not require data training. However, this method has weaknesses in that it takes a long time for computation and is very vulnerable to the addition of large amounts of data as well as low in accuracy [16]. Also, this approach is mainly memory-based, relying more on the statistical approach, that later is considered as another weakness. The above background leads to the needs for a new model (latent factor) that involves machine learning and artificial intelligence.

Since the 2006 Netflix recommender system competition, many researchers have competed to reach the highest level of accuracy in predicting ratings. In this competition, the participant who can increase the level of accuracy by more than 10 percent will be the winner. In this competition, most participants develop the model based on the latent factor approach, specifically by applying the factorization matrix [17]. There are several variants of matrix factorization implemented in a recommender system. Sarwar, Karypis, Konstan, and Riedl [18] propose a latent factor model based on collaborative modeling approach using the SVD method an abbreviation of Singular Value Decomposition. Another approach by Zhang, Wang, Ford, and Makedon [19] is a recommendation model that involves a version of matrix factorization as known NMF that is an abbreviation of Non-Negative Matrix Factorization. This model became a good model to produce rating prediction. Other researchers, Salakhutdinov and Mnih [20], develop a rating estimation method consider to applied a version of the matrix factorization method that called PMF, PMF is an abbreviation of Probabilistic Matrix Factorization. PMF model became one of a framework to generate rating estimation machine that considers only the rating element in collaborative filtering algorithm. Matrix factorization variants are very powerful in predicting product rating. However, the resulted rating prediction is inaccurate when handling sparse data. They need a supplement to product information to handle issues concerning inaccuracy.

Deep learning has tremendous achievement in several computer science research fields and has become an essential machine learning platform employed in computer science especially in NLP, NLP abbreviation of natural language processing, image processing, speech recognition, and text mining [21]. More specific Convolutional Neural Network (CNN) as a version of the neural network method that uses feed-forward processes. In our best knowledge, it is a neural network class that successful to implemented in a recommender system by applying a content-based method to generate content feature extraction, for instance, in the case of music recommendation system [11], deep learning for video recommendation based on low-level representation [22]. In this research issue, the authors exploited the side information of a product's feature by applying the CNN method to create music classification based on acoustic categories. This approach successfully dealt with semantic gap problems because the classification of textual sentences in a particular type of music often occurred in semantic gaps. The use of this model also proved effective as this model could handle the long tail problem in which unpopular music is often prevented from being recommended to the consumer. There is an alternative method to jointly side information of the user and the item to generate effective recommendation even face sparse data [23]. Incorporating information by involving deep learning applied for video recommendation on Youtube application [24].

CNN has been successfully applied in fashion e-commerce. Jaradat [13] proposes a novel approach based on the deep color classification to dig fashion model classification. He explores the relationship between color character and interest tendency in a product. Once again, the class of deep learning can increase the performance of recommendation using a content-based method to create a deep color classification of the fashion product. There have been some studies [25]–[27] concerning the development of a latent product factor that involves text sentence document of a product review from users. A product review is used for it is a widespread resource of information. Thus the information gathered is very abundant and easy to obtain. Finally, this study applied the LDA (Latent Dirichlet Allocation) approach as a representation of Bag of Word (BOW) model. These approaches successfully improved the performance of the recommendations compared to PMF [20], SVD [18], and NMF [19]. Although both models successfully improved accuracy level by embedding the side information in the form of text sentence documents from product reviews, both models failed to generate the contextually of text sentence documents. This weakness was overcome in this research by involving DCNN to explore contextual sentences in product reviews.

II. Materials and Method

Our research problem involved side information in the form of text sentence document of movie review as a representation of item latent factor. We applied a convolutional neural network, which was modified into dynamic convolutional processing. This aimed to make up for the loss of information in such approaches as the bag of the word (BOW) in the previous works' approaches. Indeed, we also involved probabilistic matrix factorization to be hybridized with the document latent factor and user information to generate the output of rating prediction of the product. Matrix factorization based on probabilistic approach has the task to calculate the incomplete rating matrix to completing the rating matrix.

This research aims to solve the sparsity data problems, as explained in section 2, and we propose hybrid approaches, including probabilistic matrix factorization and item latent factor. We expect that it will improve the results of the previous works. We adopt the existing model proposed by Wang and Blei [26], which has successfully implemented probabilistic matrix factorization and latent document factor by enhancing document product review using the LDA model. The weakness of the LDA model is that it does not catch a deep understanding of the text sentence document of a product review. We propose a novel model that applies the Convolutional Neural Network for text classification [28], [29]. We adopt this model for it ensures high accuracy in showing classification for text sentiment analysis. We also integrate the latent document factor with probabilistic matrix factorization to predict the product rating.

Indeed, to achieve the goals in exploring the contextual understanding of movie reviews, there are two important things that we have done that are capturing the surrounding word and subtle word. We involving word embedding, in this case, we applied GloVe [30] and CNN that belonging responsible for generating sentence latent factor vector. As we know that CNN output fit for classification issue [28], [31] and also this approach are not suitable to regression application, So required to incorporating with any method that common use in rating prediction. In our opinion, the matrix factorization method is the right choice to generate rating prediction. It could be also be applied with matrix factorization using the probabilistic approach [32], [20]. The basic idea on above are the things that underlie the framework that we build with the design, as shown in Figure 1 below.

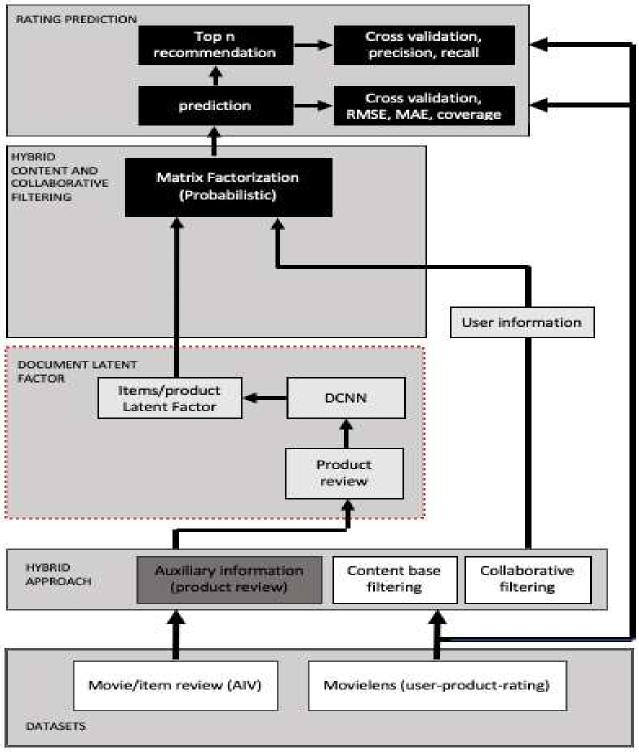

Fig. 1. Text sentence latent factor for a recommendation

The general framework of our proposed model can be seen in figure 1, where item information (sentence) play an essential role to catch product review then convert into item latent factor vector. The detail process to develop sentence item latent factor vector shown in Figure 2, where the necessary process played by word embedding and CNN.

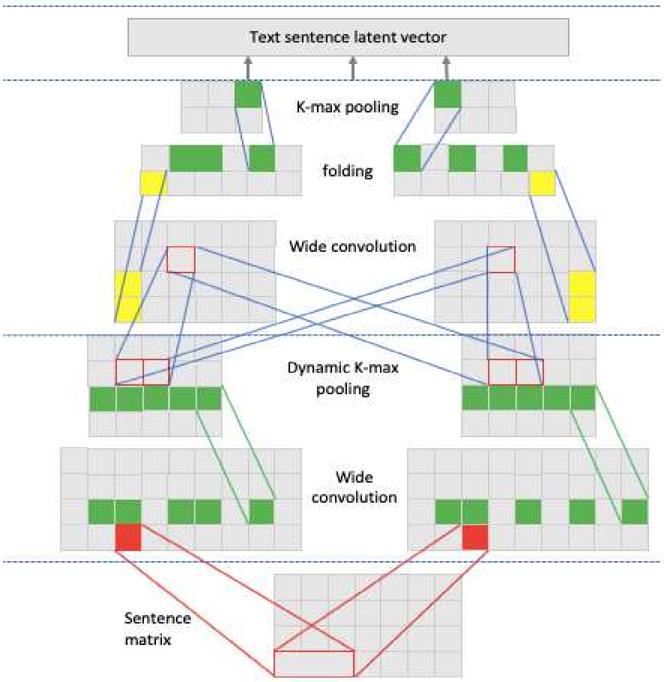

Fig. 2. DCNN latent factor vector model

Both of them having a task to catch word sequence and word surrounding in the sentences. Word embedding is a modern pre-processing method to extract similarity length of the word in a sentence of item document.

A. Word Embedding and CNN

According to the underlying problem of above, we proposed a new model considering CNN and Word embedding to solve the problem in contextually sentence understanding of item properties due commonly properties belonging of item feature is in the form of sentences. The model that we applied this method named DCNN that abbreviation of Dynamic Convolutional Neural Network. The model designed to acquire sentence latent factor vector uses the document belonging product review. Thus, the strategy to achieve the objective, we prepare to generate sentence item latent factor vector as follow:

1) Inputting Layer Process: This layer requires to take responsibility for managing the input data from the product review of the dataset. This layer has functions to process inputs in the form of a collection of reviews on the products, in this case, taking an example of a report of a film. The information in the form of sentences cannot be directly executed for the training data process. Some of the important processes are like removing stop word, eliminating conjunctions, returning words to basic words. This is important to do to facilitate the computational process but does not reduce the representation of the meaning of a sentence.

2) Embedding Layer Process: The embedding layer has responsible for converting a column of the word in the sentence to the form of a numerical number, then storing the numeric into the table matrix. After that, the process continues into a convolutional layer step. An example, if we have a document of which the number of words is l, we can concatenate a vector representation of every word then taken entire into table matrix that concern word order of the original sentences. Word vector was acquired from word embedding process where p represents word embedding, and w is representation raw of the word.

3) Convolution Layer process: the convolutional layer have responsible to pull out sentences semantic from items properties. This research considers involving movie product review that generates from the imdb portal. This experiment implemented convolutional to manage the sentence. We exploit $c_i^j \in R$ as a representation of contextual properties and $w_c^j \in \mathbb{R}^{p \times ws}$ to initialized share weight of the word in the sentences and windows measure to obtain the score of surrounding word.

$$c_i^j = f(w_c^j * D(:, i:(i+ws-1)) + b_c^j) \tag{1}$$

Which $*$ is a symbol of convolutional step, $b_c^j \in R$ representation of bias for W_c^j also $f()$ is a representation of the activation function? Several various activation functions are ReLU (an abbreviation of Rectified Linear Unit), Sigmoid and Tanh. In this case, decided to apply ReLU to finalize such vanishing gradient case. Thus, to generate contextual characteristic vector given by $c^j \in \mathbb{R}^{l-ws+1}$ from document W_c^j enhancing using:

$$c^j = [c_1^j, c_2^j, c_3^j, c_4^j, c_5^j ...., c_{l-ws+1}^j] \tag{2}$$

4) Pooling Layer Process: This layer belonging responsible for digging contextual characteristic from the convolutional process also collect feature-length vector of sentences document. Since the convolutional process complete to finish, then generate a contextual characteristic vector by NC as a representation of the sentencing document. We consider to applied max pooling method to enhance previous work in the convolutional layer, Max pooling layer calculated by given:

$$df = [\max(c^1), \max(c^2), ..., \max(c^j), ..., \max(c^{n_c})] \tag{3}$$

Where c^j representation of the contextual characteristic vector that obtains from the variable length of the sentencing document given by $l - ws + 1$.

5) Output Layer Process: Finally, the output layer responsible for receiving the process in the previous work became the output in the term of item latent factor vector by applied nonlinear model given by equation as follows:

$$s = \tanh(W_{f2}\left\{\tanh(W_{f1} d_f + b_{f1})\right\} + b_{f2}) \tag{4}$$

Where $w_{f1} \in \mathbb{R}^{f \times n_c}$ and $w_{f2} \in \mathbb{R}^{k \times f}$ as estimation matrix representation. Since $b_{f1} \in \mathbb{R}^f$ and $b_{f2} \in \mathbb{R}^k$ is representative the bias vector for $w_{f1}$and w_{f2} with $s \in \mathbb{R}^k$. Finally, after all of the process has finished, the last process is generating sentence document item latent factor vector of every document with the formula as follow:

$$s_j = dcnn(W, Y_j) \tag{5}$$

Which *W* indicates bias factor and weight factor in avoiding chaos condition, text sentence of document representation indicated by Y_j that belonging item *j*. The last is to represent text sentences document latent factor vector represent $s_j$that belonging to item *j*.

B. Probabilistic MF

In this research, we consider adopting the method that similar to Salakhutdinov and Mnih [16]. We have integrated text sentence document latent factor into matrix factorization, and further, we modified this approach by referring to Kim's method [28], which performed the highest achievement in text sentiment classifier. We had incorporated Probabilistic Matrix Factorization (PMF) approach and item latent factor representation from text sentence document in our model. It aimed to generate a text sentence document latent factor called the Dynamic Convolutional Neural Network (DCNN). When assume belonging a matrix with *N* represent of users and *M*, represent of items. Thus, the rating can be predicted by $R \in \mathbb{R}^{N \times M}$ matrix.

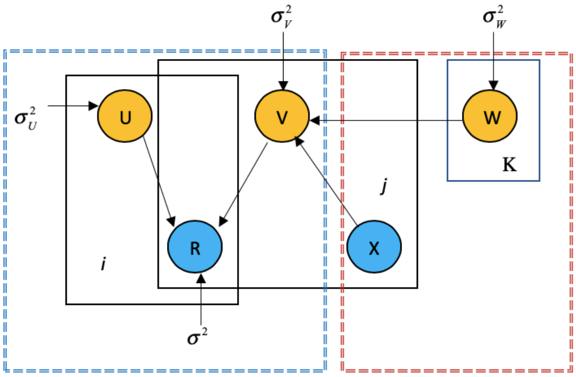

Fig. 3. Incorporating sentence latent factor and matrix

Then, our objective is to find the user and item latent models ($U \in \mathbb{R}^{N\times M}$ and $V \in \mathbb{R}^{N\times M}$) of which product ($U^T V$) reconfigures the rating matrix *R*. Based on the probabilistic method that implements normal distribution, the learn method to generate rating prediction can be acquired by giving:

$$p(R|U,V,\sigma^2) = \prod_i^N \prod_j^M N(r_{ij}|u_i^T v_j,\sigma^2)^{I_{ij}} \quad (6)$$

Which $N(x|\mu,\sigma^2)$ is a representation of probability density function that very commonly applied in normal distribution on the Gaussian method. Symbol of μ is represent of mean value and σ^2 represent of variance value. While I_{ij} is represent of the indicator function. To develop the user, latent factor model, we involve exploiting a zero-mean spherical Gaussian prior as popular conventional prior on latent user model. In this process, we use variance σ_U^2.

$$p(U|\sigma_U^2) = \prod_i^N N(u_i|0,\sigma_U^2 I) \quad (7)$$

It is a very different way from the traditional item latent factor based PMF, and thus we consider three elements: 1) Symbol *W* represents of internal weight in our method, 2) X_j reflect of the sentence of the document that belonging item *j*, also 3) reflect epsilon variable factor come from Gaussian noise. Concern to maximization of the item latent method to predict the rating, we applied the following to the formula to generate the final item latent model:

$$\in j \sim N(0,\sigma_V^2 I) \quad (8)$$

where weight w_k is in *W*, and we use a traditional zero mean spherical Gaussian prior is as in the following equation:

$$p(W|\sigma_W^2) = \prod_k N(w_k|0,\sigma_W^2) \quad (9)$$

Meanwhile, the distribution of latent item model is denoted by:

$$p(V|W,X,\sigma_V^2) = \prod_j^M N(v_j|cnn(W,X_j),\sigma_V^2 I) \quad (9)$$

Where *X* reflects a group of text sentence review document belonging items. A review of product latent vector acquired by the model designed on previous section namely DCNN as the abbreviation of Dynamic Convolutional Neural Network that adopts mean come from Gaussian distribution noise belonging-product that exploited became variance value own Gaussian distribution to unite the dispute among DCNN and Probabilistic Matrix to incorporate between item review the document and rating variable.

This study designs the framework to eliminate sparse data especially for user rating in case of collaborative method to obtain product recommendation that involve deep learning machine to find the relationship among document latent factor of the product It is commonly acknowledged that the previous work include only matrix factorization fails to address sparse data issue in user rating, leading to inaccurate recommendations. [17]. This research was classified into laboratory scale experiment applying e-commerce public datasets. We employed MovieLens' dataset that was applied in many e-commerce recommender system studies. The detailed specification of this dataset referred to the work of Harper and Konstan [33]. The additional information on product review was obtained from Amazon Information Video (AIV). The detailed characteristics of the datasets are shown in Table 2.

TABLE 1
LIST OF HARDWARE AND TOOLS

Hardware/library	Specification
CPU	Xeon 4 core, 2.4
RAM	16 Gb
GPU/VGA	Nvidia GTX 1001
Keras	Deep NN
Tensor Flow	Deep NN
Anaconda	Python web interface
Python 3.0	Programming language
NLTK	NLP processing
Cuda	GPU CNN
Mathplotlib	Visualization data

This research focus to enhancing text sentence document from the item, and converting the document becomes item latent factor vector; We considered several datasets referring [33] and AIV as references of the sources for a product review for the movie. We have detailed characteristic dataset that we used shown in table 3.

TABLE II
DATASET SPECIFICATION

Datasets	Users	Products	Rating	Sparse (%)
ML-100k	943	1.682	100,000	6.3%
ML-1M	6.040	3.900	1.000.209	4.2%
ML-10M	71.567	10.681	10.000.054	1.3%
AIV	29.757	15.149	135.188	0.030%

C. Hardware and Library

Our experiment relied on software and hardware. The software employed included Python programming language including some libraries like Tensor flow, Keras, Nltk to implement deep learning application, while the hardware used was Nvidia GPU with GTX 1001 to compute CNN and

Intel processors Xeon Quad Core 2.4 GHz. The detailed specification of our tools is shown in Table 1.

D. Dataset Used

The experiment in this research aims to demonstrate the firm of our model to calculate rating estimation; we implemented two e-commerce dataset collect from 2 real e-commerce company including Movilens [33] and Amazon [34] mixed with IMDB. These datasets contained consumer' explicit ratings for products on rating star 1-5. Dataset collected from Amazon include item review documents. The Amazon dataset relevant to embedded due to own dataset MovieLensdo not include feedback by customers, we produced the feedback document by developing a connection with IMDB web service. As what [26] and [25] did in their study, we used pre-processed description document including overall dataset as follows: (1) Design the document length of upper value maximize in 300 words. It is normally according to some literature review in previous work. (2) Delete unused stop word to reduce high computation. (3) Counting every word in the sentence use *tf-idf* method. (4) Remove corpus with unique stop word belonging a document frequency more than 0.5. (5) Collecting the most 7,000 different words that categorical vocabulary. (6) Delete every non-vocabulary expression in raw table documents.

Accordingly, the mean number of the word for every document is 96.04 own ml-1m by MovieLens, 93.07 belonging by ml-10m also on Movielens and 89.10 belonging IMBD and Amazon. We also deleted the member of the item that has no review sentences entire the table in datasets. Specifically, on Amazon side data, we removed the user's data of which rating was less than three total number rated product. The result of the statistical calculation in every data demonstrates three model datasets belonging distinction feature, as presented in Table 2. Finally, while several users have deleted from the table datasets on the pre-processing phase, it was quite sparse data when calculating to another data.

E. Evaluation Metrics

Accordingly, to prove and evaluate the recommendation model that we have proposed, we consider using the RMSE evaluation method. It is one of most popular evaluation approach to measuring the level of accuracy in rating prediction, especially on the collaborative filtering research field, is RMSE [4], it is an abbreviation of root mean squared error. The basic idea of RMSE that measure of the difference between actual rating and prediction rating. It uses a representation of standard deviation in this practical implementation. To ensure the best performance of our model, we carried out an evaluation metric by using RMSE that is straight connected to the objective function of the output in rating prediction.

$$RMSE = \sqrt{\frac{\sum_{i,j}^{N,M}(r_{ij} - \hat{r}_{ij})^2}{\#rating}} \tag{10}$$

III. RESULT AND DISCUSSION

The results of this study are presented in the following graphics and diagrams. Each result from the data training is elaborated concisely. Based on our experiments by involving the Movielens 1 million (ML-1M) and Movie review (AIV) datasets and considering several ratios (20:80, 30:70, 40:60) for training data, the model we have proposed has successfully generated rating prediction. The detailed results of the training data evaluation using RMSE with three ratio compositions are shown in Figure 4.

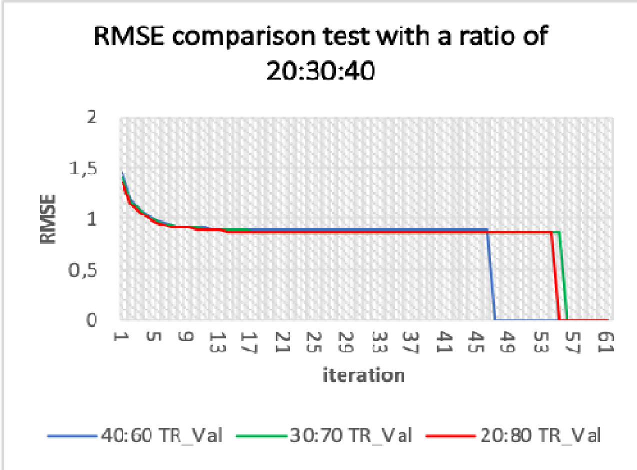

Fig 4. Results of RMSE test on various training data ratios

Training data with a ratio of 40:60 could produce the most accurate rating predictions with the most efficient time. The resulted rating prediction of the product based on the RMSE average level was 0.9212935, meaning that it had the lowest accuracy over the other training data tests in the RMSE evaluation. For the 30:70 ration, our model could yield a more accurate rating prediction than the existing model with an average score of 0.8720675. However, it required much more time to complete the process. The best rating prediction was obtained on a data ratio of 20:80, in which the average result was 0.8617935 based on the RMSE evaluation metric. However, it still took a lot of time to process the data compared to the other training data. In this model testing scenario, if the training evaluation does not increase, or it has a saturation value, the training data will be stopped automatically.

The results of the second test were compared with those of previous research on PMF, CDL, and CTR. PMF only involves user items and ratings using a matrix factorization approach without involving additional information. In our experiment, we combined the probabilistic matrix factorization and at the same time, added information from items in the form of latent factor representation items from product reviews. The addition of information that we did could make rating predictions more accurate. Thus the model we proposed had better performance than the existing models that only relied on probabilistic matrix factorization.

The comparison results are then contrasted with probabilistic matrix factorization models that involved additional information in the form of product reviews, namely CTR and CDL. The item latent factor approach, based on text sentence representation, employed the LDA model. The result of the comparison between CTR and CDL in the 3 training ratio datasets was 30:70. DCNN gained the highest accuracy rate of 0.8720675 compared to that of the previous works: 0.9639 (CTR) and 0.9685 (CDL). The detailed results of this comparison are presented in Figure 8.

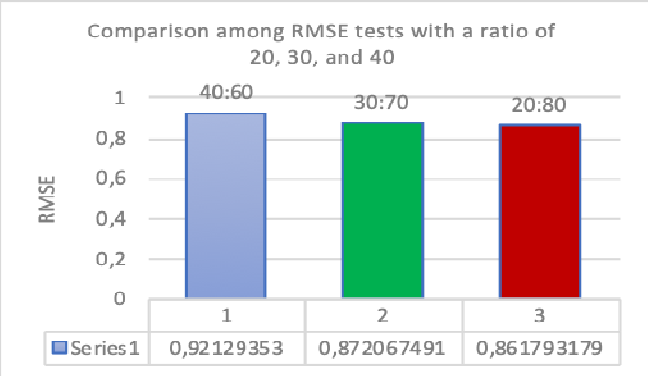

Fig 5. Comparison among RMSE tests with a ratio of 20, 30, and 40

In the next comparison, at a data ratio of 20:80, our model could produce a better achievement using the RMSE evaluation to measure the level of accuracy compared to the existing model. The detailed completion of the RMSE test is shown in Figure 9 below. Our DCNN model performed better than the PMF, CDL, and CTR did.

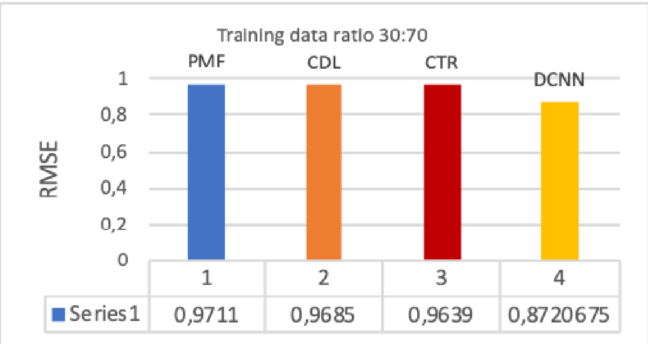

Fig 6. Comparison between RMSE and the previous works (ratio 30:70)

In the next comparison on data training with a ratio of 40:60, our model also could produce a better level of accuracy than the existing model. The detailed results of the RMSE test are shown in Figure 10. Our DCNN model outperformed PMF, CDL, and CTR in terms of the obtained accuracy.

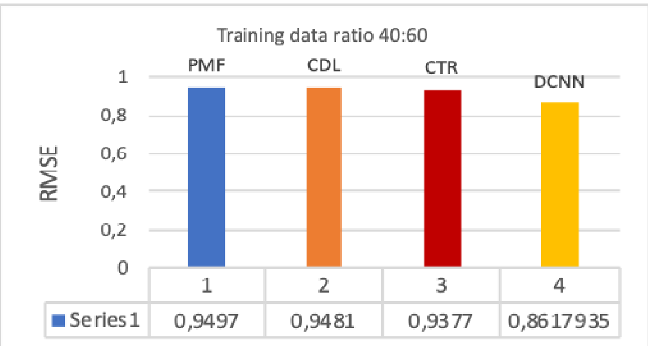

Fig 7. Comparison between RMSE and the previous works (ratio 40:60)

In the last comparison on data training ratio 20:80, our model also could produce a better level of accuracy than the existing model. The detailed results of the RMSE test are shown in Figure 11. Using the RMSE, the accuracy obtained by our DCNN model outperformed that of PMF, CDL, and CTR.

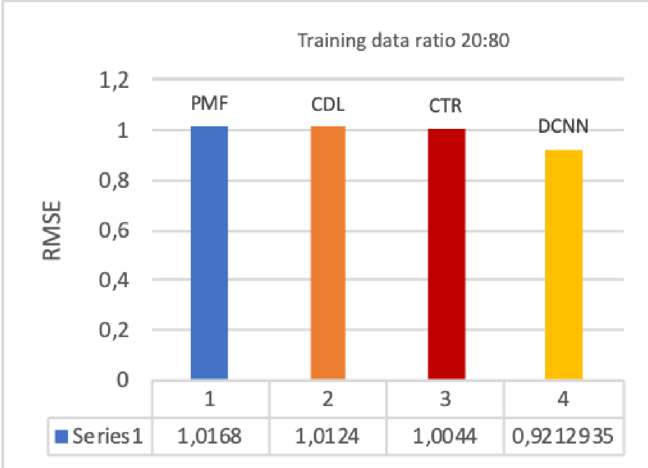

Fig 8. Comparison between RMSE and the previous works (ratio 20:80)

IV. Conclusion

Our experiment, which involved deep learning (DCNN) and word embedding approach in developing a novel model to dig contextual meaning into item latent factor representation, had been successfully implemented. Improved understanding of text sentence documents about product reviews from customers affects the effectiveness of rating predictions. It has a present an essential role in obtaining a much better result in our work with the DCNN model and word embedding. Our work has achieved better results than the others' because our model can capture sentence document of product review that have a subtle connection between one word and another around it. The effects of the convolutional dimension with the K-Max pooling and the wide convolutional method greatly influence the contextual sensitivity of words in a sentence, so that product ratings will be more accurate if evaluated using RMSE.

Acknowledgment

We would like to thank for some institution that is often supporting us to conduct research that are Universitas Amikom, Universiti Teknikal Malaysia Melaka, PT. Time Excelindo.

References

[1] C. G. Uribe, N. Hunt, and N. Inc, "The Netflix Recommender System: Algorithms, Business Value, and Innovation," *ACM Trans. Inf. Syst.*, vol. 6, no. 4, p. 19, 2015.

[2] J. Davidson, B. Liebald, J. Liu, P. Nandy, and T. Van Vleet, "The YouTube Video Recommendation System," in *ACM Recsys 2010*, 2010, no. August, pp. 293–296.

[3] J. Ben Schafer, J. Konstan, and J. Riedl, "E-commerce recommendation applications," *Appl. Data Min. to Electron. …*, pp. 115–153, 2001.

[4] Hanafi, N. Suryana, and A. Sammad, "An understanding and approach solution for cold start problem associated with

recommender system : a Literature Review," *J. Theor. Appl. Inf. Technol.*, vol. 96, no. 09, pp. 2677–2695, 2018.
[5] E. Çano and M. Morisio, "Hybrid recommender systems: A systematic literature review," *Intell. Data Anal.*, vol. 21, no. 6, pp. 1487–1524, 2017.
[6] M. Elahi, F. Ricci, and N. Rubens, "A survey of active learning in collaborative filtering recommender systems," *Comput. Sci. Rev.*, vol. 20, pp. 29–50, 2016.
[7] F. Belletti, K. Lakshmanan, W. Krichene, Y.-F. Chen, and J. Anderson, "Scalable Realistic Recommendation Datasets Through Fractal Expansions," *arXiv Prepr. arXiv1901.08910*, 2019.
[8] C. C. Aggarwal, *Recommender systems: The Textbook*. London: Springer International Publishin, Switzerland, 2016.
[9] D. Kotkov, S. Wang, and J. Veijalainen, "A survey of serendipity in recommender systems," *Knowledge-Based Syst.*, vol. 111, no. August, pp. 180–192, 2016.
[10] J. Wei, J. He, K. Chen, Y. Zhou and Z. Tang, "Collaborative filtering and deep learning based recommendation system for cold start items," *Expert Syst. Appl.*, vol. 69, pp. 1339–1351, 2017.
[11] A. van den Oord, S. Dieleman, and B. Schrauwen, "Deep Content-Based Music Recommendation," *NIPS*, pp. 2643–2651, 2013.
[12] X. Wang and Y. Wang, "Improving Content-based and Hybrid Music Recommendation Using Deep Learning," *Proc. 22Nd ACM Int. Conf. Multimedia.*, pp. 627–636, 2014.
[13] S. Jaradat, "Deep Cross-Domain Fashion Recommendation," *Proc. Elev. ACM Conf. Recomm. Syst. - RecSys '17*, pp. 407–410, 2017.
[14] K. Park, J. Lee, and J. Choi, "Deep Neural Networks for News Recommendations," *CIKM'17, November. 6-10, 2017, Singapore*, 2017.
[15] X. Wang *et al.*, "Dynamic Attention Deep Model for Article Recommendation by Learning Human Editors' Demonstration," *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. - KDD '17*, pp. 2051–2059, 2017.
[16] F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*, 2nd ed. London: Springer New York Heidelberg Dordrecht London, 2015.
[17] Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," *IEEE*, vol. 40, no. 8, pp. 42–49, 2009.
[18] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Incremental Singular Value Decomposition Algorithms for Highly Scalable Recommender Systems," *Am. Lab.*, vol. 37, no. 10, p. 4, 2005.
[19] S. Zhang, W. Wang, J. Ford, and F. Makedon, "Learning from Incomplete Ratings Using Non-negative Matrix Factorization," *Proc. 2006 SIAM Int. Conf. Data Min.*, pp. 549–553, 2006.
[20] R. Salakhutdinov and A. Mnih, "Probabilistic Matrix Factorization." *Proc. Adv. Neural Inf. Process. Syst. 20 (NIPS 07)*, pp. 1257–1264, 2007.
[21] L. Zheng, *A Survey and Critique of Deep Learning on Recommender Systems*, 1st ed., no. September. Chicago: University Of Illinois At Chicago, 2016.
[22] R. J. R. Filho, J. Wehrmann, and R. C. Barros, "Leveraging Deep Visual Features for Content-based Movie Recommender Systems," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2017-May, pp. 604–611, 2017.
[23] X. Dong, L. Yu, Z. Wu, Y. Sun, L. Yuan, and F. Zhang, "A Hybrid Collaborative Filtering Model with Deep Structure for Recommender Systems," *Aaai*, pp. 1309–1315, 2017.
[24] P. Covington, J. Adams, and E. Sargin, "Deep Neural Networks for YouTube Recommendations," *Proc. 10th ACM Conf. Recomm. Syst. - RecSys '16*, pp. 191–198, 2016.
[25] H. Wang and D. Yeung, "Collaborative Deep Learning for Recommender Systems arXiv : 1409. 2944v1 [cs . LG] 10 Sep 2014," no. July 2015, 2014.
[26] C. Wang and D. M. Blei, "Collaborative Topic Modeling for Recommending Scientific Articles," *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discov. data Min. - KDD '11*, p. 448, 2011.
[27] L. Zheng, V. Noroozi, and P. S. Yu, "Joint Deep Modeling of Users and Items Using Reviews for Recommendation," in *WSDM 2017*, 2017, no. February, pp. 425–434.
[28] 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.
[29] D. Kim, C. Park, J. Oh, S. Lee, and H. Yu, "Convolutional Matrix Factorization for Document Context-Aware Recommendation," *Proc. 10th ACM Conf. Recomm. Syst. - RecSys '16*, pp. 233–240, 2016.
[30] J. Pennington, R. Socher, and C. Manning, "Glove: Global Vectors for Word Representation," *Proc. 2014 Conf. Empir. Methods Nat. Lang. Process.*, pp. 1532–1543, 2014.
[31] E. Çano and M. Morisio, "A deep learning architecture for sentiment analysis," *Proc. Int. Conf. Geoinformatics Data Anal. - ICGDA '18*, no. April, pp. 122–126, 2018.
[32] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative Deep Learning for Recommender Systems," in *KDD conference*, 2015, pp. 1235–1244.
[33] F. M. Harper and J. A. Konstan, "The MovieLens Datasets: History and Context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 19:1--19:19, 2015.
[34] J. McAuley and J. Leskovec, "Hidden factors and hidden topics," *Proc. 7th ACM Conf. Recomm. Syst. - RecSys '13*, pp. 165–172, 2013.