Deep convolutional neural networks for product recommendation

N. Lakshmipathi Anantha[1,*], Bhanu Prakash Battula[2]

1. Acharya Nagarjuna University, Department of IT, VFSTR Deemed to be University, Vadlamudi, India

2. Department of CSE, Thirumala Engineering College, Jonnalagadda (V), India

anlakshmipathi@gmail.com

ABSTRACT. *Recommender Systems are big breakthrough for the web enabled systems as Recommender Systems have the capability to analyze the behavior patterns of the user. And these systems are accomplishing the task of recommending the products the users are interested in. Existed models grabbing the insights of the users and items patterns will give satisfactory results to the users. This paper uses pretrained models to extract the knowledge from the data using the concept of transfer learning. Our models use the knowledge of pre-trained models to extract patterns between users and items. To facilitate this objective, in this paper we presented our approach to generate recommendations in two phases. In the Classification phase, classification of product images and its experimental analysis following, the Ranking phase to rank the product images to the user and its experimental analysis are discussed. The result analysis discussed in this paper achieved promising results.*

RÉSUMÉ. *Les systèmes de recommandation constituent une percée majeure pour les systèmes en ligne car ils permettent d'analyser le mode de comportements de l'utilisateur. Et ces systèmes accomplissent la tâche de recommander les produits qui intéressent les utilisateurs. Les modèles existants capturant les informations des utilisateurs et les modèles d'éléments qui donneront des résultats satisfaisants aux utilisateurs. Cet article utilise des modèles pré-entraînés pour extraire les connaissances à partir des données en utilisant le concept d'apprentissage par transfert. Nos modèles utilisent la connaissance des modèles pré-entraînés pour extraire des motifs entre les utilisateurs et les produits. Pour faciliter cet objectif, dans cet article, nous avons présenté notre approche pour générer des recommandations en deux phases. Au cours de la phase de classification, la classification des images de produits et son analyse expérimentale sont ensuite décrites. Et puis, la phase de classement pour classer les images de produit pour l'utilisateur et son analyse expérimentale sont discutées. L'analyse des résultats présentée dans cet article a donné des suites prometteurs.*

KEYWORDS: *recommender system, convolutional neural network, content-based filtering, ranking.*

MOTS-CLÉS: *système de recommandation, réseau de neurones de convolution, filtrage basé sur le contenu, classement.*

DOI:10.3166/ISI.23.6.161-172

1. Introduction

Every one of us experiencing flood of information with the fast development of the world wide web. It is estimated that WWW contains 4.44 billion pages and 146.77 million pages are indexed in Dutch Indexed Web. Based on these statistics every day the number of web pages are indexed in WWW. When a user browses from the large sized WWW it is impossible to locate their content from the huge data which is resulted output of searching.

For efficient filtering of content, we must have a filtering technique which should bring user interested contents. This task is done by Recommender Systems. It is also called as Recommendation Systems. It is mainly classified into 3 categories. They are as follows. Content based filtering, collaborative filtering and hybrid Approach. Recommender Systems takes mainly two types of data. The first type is the user-item interactions like ratings given by the users or buying pattern or buying behavior. The second type is the attribute information about the users and items such as textual profiles or relevant keywords. Methods process the first approach is referred to as Collaborative Filtering and the methods process the second approach is referred to as Content-based Filtering. Pros of both Content-based Filtering and Collaborative Filtering are referred as Hybrid approach.

Lot of research has been carried out on the Recommender Systems. Now a days Deep Learning, a sub field of Machine Learning is offering superior performance in computer vision and Natural Language processing and other application areas. We used Convolutional Neural Network one of the techniques of the deep leaning in this paper for classification products in classification phase. After Classification Phase, Pairwise Ranking is used to rank the products in ranking phase. Ranking the products is one of the important parts in Recommender Systems. In this paper the proposed ranking model achieved good results.

This paper is organized as follows: section-II describes the related work about Traditional Recommender Systems, Deep Learning based Recommender Systems and Ranking approaches. Section-III describes about Classification Phase and its Result Analysis. Section-IV describes about Ranking Phase and its Result Analysis. Finally, section-V is about the conclusion of the paper.

2. Related work

2.1. Traditional recommender systems

Content-based filtering (Meteren and Someren, 2000) generates recommendations based on the activities a user has performed in his account. The advantages of Content-based filtering are, it can generate recommendation even the database does not have user preferences and can generate recommendations more accurately if the user's preferences are changed. Content based filtering is used in the following systems. They are as follows. News Dude, LIBRA, etc. The pitfalls of content-based filtering are sparsity, limited content analysis. Collaborative filtering

(Schafer *et al.*, 2007) is very popular technique in Recommender Systems. The main source of Collaborative filtering is ratings. Collaborative filtering is classified in to 2 types. They are as follows. Memory-based techniques and Model-based Techniques. In Memory-based techniques (Breese *et al.*, 1998) entire data must be used for training and for giving recommendations. While, Model-based techniques (Cheng, & Hurley, 2009) not require entire data for training. Part of the data can be used for training. Based on that knowledge, it can perform recommendations. The advantages of Collaborative filtering are it can recommend items which are not in the profile of the user. The main disadvantages of Collaborative filtering are Cold-start problem, data sparsity, scalability. Collaborative filtering is used in the following systems. They are as follows. Ringo, Grouplens, Amazon.com, etc., The pitfalls of Collaborative filtering are cold start problem, scalability and sparsity. To overcome the pitfalls in content based filtering and collaborative filtering Hybrid Approach came into picture. Hybrid Approach (Hummel *et al.*, 2007) combines one more approach in different way to improve the results. They can be classified based on their operations into feature-combination hybrid,feature-augmented hybrid, switching hybrid, weighted hybrid, mixed hybrid, cascade hybrid and meta-level hybrid.

2.2. Deep learning based recommender systems

Deep Learningis the combination of Artificial Neural Network (ANN) and Machine Learning (ML). We have different approaches for different problems in deep learning. The simple approach is Multilayer perceptron (MLP). MLP is set of layers. It has one input layer and one output layer. In between input and out layer, other layer used for processing the data. In MLP, the data forwarded from input layer to the next layers. Finally, data received at the output layer. Neural Collaborative Filtering (He *et al.*, 2017) is framework based on MLP try to capture the non-linearity between Users and Products. Auto-encoder (Baldi, 2012) is another approach is unsupervised approach uses backpropagation. Auto-encoders contains encoder and decoder. Encoders role is to transform the data into easy and simple format. Decoders role is to decode back the original data. AutoRec (Sedhain *et al.*, 2015) is Framework based on Autoencoders tries to reconstruct the input. It offers both Item-based Auto-rec and User-based AutoRec. Restricted Boltzmann Machine (RBM) is another approach, it uses the concept of hidden layers. It has both visible and hidden layers. In RBM, input is provided at input layer. the data forwarded to the next layers. Output received in the output layer. each node in a layer doesn't have connection with other nodes (inter-node connections). This is the first model which is built on top of Deep Learning. Netflix Prize Contest is the evidence for this model is offering best performance. Restricted Boltzmann Machine Collaborative filtering (RBM-CF) (Georgiev and Nakov, 2013) is a framework offers both user-based and item-based recommendations using RBM.

Convolutional Neural Network also grabbing researchers' interest towards Computer vision and Recurrent Neural Networks are using in Natural Language Processing areas (Szegedy *et al.*, 2015). In this paper, in Classification Phase

Convolutional Neural Networks are used to classify the images. After CNN (Classification Phase), we proposed a ranking model (Ranking Phase) uses pairwise ranking approach and provides top-N product recommendations.

2.3. Ranking approaches

Learning to rank is popular technique used to rank the products/web pages. Recommender Systems deals lots of products. To perform recommendation, we need to rank the products. Learning to rank has mainly the following 3 approaches. They are listwise approach, pairwise approach and pointwise approach. These approaches fall under Neural Net techniques, Boosting Techniques, SVM techniques and others. The following table 1 classifies the different available learn to rank approaches.

Table 1. Classification of learn to rank approaches

SVM		Boosting	NeuralNet	Others
Pointwise	OC SVM	McRank		Prank, Subdet Ranking
Pointwise	Ranking SVM IR SVM	RankBoost GBRank LambdaMART	RankNet Frank LambdaRank	
Listwise	SVM MAP PermuRank	AdaRank	ListNet ListMLE	SoftRank AppRank

Pointwise approach processes a single product at a time. It takes single product and trains the model on it. The model predicts the relevance for the current product. The final ranking is achieved by simply sorting the result list by these product scores. *Pairwise Approach* process a pair of products at a time. Based on the scores this approach orders the products. *Listwise Approach* process entire product list and comes with the optimal ordered list of products.

3. Classification phase

In this phase we use convolutional Neural Network to classify the images. General diagram of a convolutional network is depicted in the fig.1. Our proposed model uses pre-trained Convolution Neural Network comprises of stacked Convolution layer, Pooling layers and Fully Connected Layers. Our approach uses the pre-trained architectures weights using transfer learning approach. Training the networks with large datasets from the starting takes days. In this paper the proposed approach uses ResNet-50 pretrained model for classification of labels. ResNet-50 model is trained on image data of ImageNet (Lecun *et al.*, 1998). ResNet-50 is a big network with 50 layers.

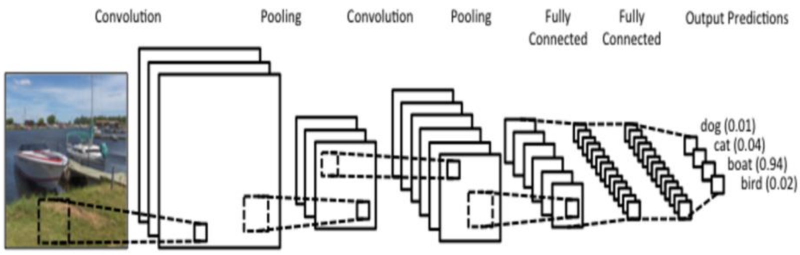


Figure 1. Convolutional Neural Network

Convolutional Neural Network is like regular Neural Networks. These networks consist of neurons which has learnable weights and biases. These weights and biases are given as input to the neurons and performs weighted sum over them and passes output to the next layers.

Convolutional Neural Networks are used to train on Images. Regular Neural Networks can be used to process small size images but to process quality images it is better to use convolutional neural networks. For example, if an image of size 32x32x3(width, height and color) produces 3072 weights. Similarly, if an image of size 200x200x3 produces 120,000 weights. To process quality images Convolution Neural Network uses Pooling layer, which reduces the weights without losing the information.

Convolution layer is the first part in convolution neural network. It extracts the features of an image. We take Convolution or kernel or filter and slide it over the image. The result we obtained after a convolution applied on the image is called activation map. Generally, five to six different filters are convolved on the image. We apply convolution on the images to identify the geometrical shapes, edge detection, blur areas and sharp areas. When the image sizes are too large to reduce the number of parameters **Pooling layers** are used. pooling techniques are classified into 3 types. They are as follows. Max Pooling, Average Pooling and sum pooling.

3.2. Pre-trained models:

LeNet-2010: This model (Baldi, 2012) is developed by Yann LeCun. This model is a shallow architecture, employed 2 Convolutional Layers and 2 Fully Connected Layers. This model uses 5x5 filter. Takes 32x32x1 size image as input. Model uses 60 Thousand parameters.

AlexNet-2012: This model (Krizhevsky *et al.*, 2012) is developed by Alex Krizhevsky. AlexNet was designed by SuperVision Group. This model is 8 layered architecture, employed 5-Convoultional Layers and 3- Fully Connected Layers. This model was trained on 15 million images with 1000 categories. Takes 227x227x3 size image as input.

ZFNet-2013: This model (Zeiler, and Fergus, 2014) is developed by Mathew Zeiler and Rob Fergus. This model is 8 layered architecture and has 4 Convolutional layers and 2 Fully Connected Layers. Takes 224x224x3 size image as input.

VGGNet-2014: This model (Szegedy *et al.*, 2015) is developed by Simonyan and Zisserman. this model is 19 layered architecture and employed 5 Convolutional Layers and 3 Fully Connected Layers. Takes 224x224x3 size image as input.

GoogleNet-2014: This model (Simonyan and Zisserman, 2014) is developed by Google. The other name for this model is Inception. This model is a 22 layered architecture. The number of parameters reduced from 60 Million (AlexNet) to 4 Million. It has employed different sized Convolutional Layers.

ResNet-2015: It is also called as Residual Neural Network (He *et al.*, 2016). This model is developed by Kaiming He. This model is 152 layered architecture. This model was trained on 15 million images with 1000 categories.

In E-commerce perspective/scenario, users face difficulty in choosing the products/items from the large scale of alternatives. These E-commerce companies maintains lot of products and their meta-data (features, reviews, ratings). When the user choosing an item, item category plays vital role. Sometimes the items may fall under more than one category. Hence, we present a CNN based product image classification approach whichextracts category features. Then, the feature map extracted from each category is used as potential information to generate ranked products.

plenty of image classification models are available. In general, knowledge gained from the model preserved in the form of weights. To train a model, model takes days to get trained on the huge data. To overcome this problem, we use the concept of Transfer Learning in pre-trained models. Transfer learning is a concept used to apply the knowledge gained on one field on another field. We take the weights from the pre-trained model as starting point and train the model on our data. Our model uses ResNet-50 architecture. ResNet-50 contains 50 layers trained on the millions of labelled data of ImageNet. In this way we perform the classification process. After this phase we perform ranking of the products.

3.3. Result analysis:

The effectiveness of the proposed approach, for generating items classification is discussed here. The dataset for the evaluation process is grabbed from the Movie Dataset (Montaleão Brum Alves, 2017). Our approach uses ResNet-50 for classifying the items category using the images of the items. ResNet won first prize in ILSVRC 2015 classification competition. This architecture has 5 blocks of convolution layers and each block is three layers deep. The proposed approach uses The Movie dataset which comprises of 6000 records. The data set contains metadata about movies and its item types. Each image in the dataset belongs to one of the categories(genre). Each Image falls under one of the 20 genres. The following diagram depicts the number of instances of movie genre of the dataset.

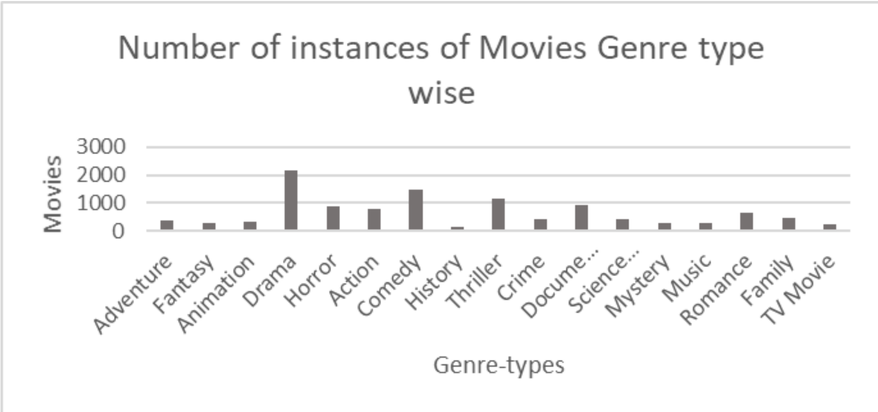
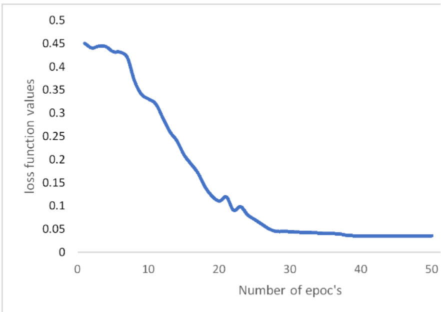
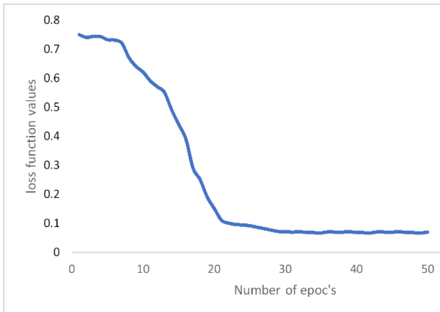


Figure 2. The instances of genre-types

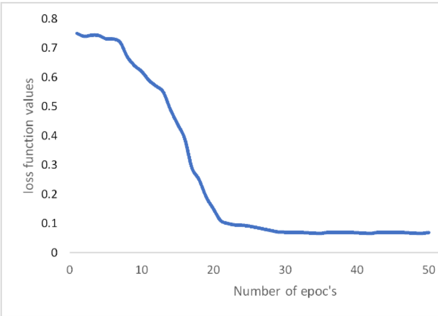
By observing the instances of genre type, the items belonging to these categories are classified and performance of the model is depicted in the figure. The dataset is partitioned into training set and test set. The model is trained on the training set and tested over the test set using precision recall metric obtained satisfactory results. In classification phase, precision means the ratio of intersection of relevant products and retrieved products to retrieved products generated from that genre type. Recall is the ratio of intersection of relevant products and retrieved products to relevant products generated from that genre type.



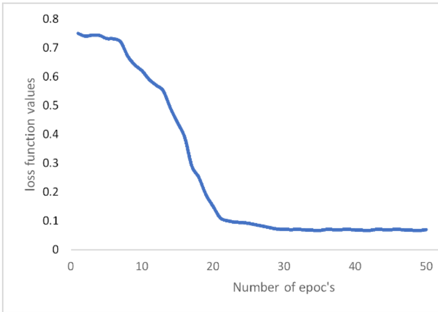
(a).Loss values on Drama



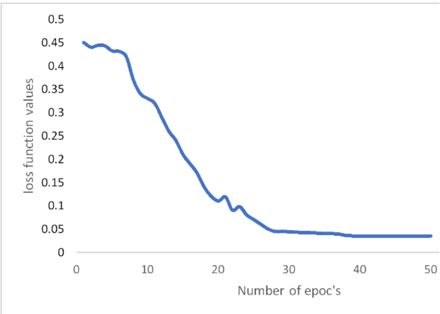
(b).Loss values on Thriller



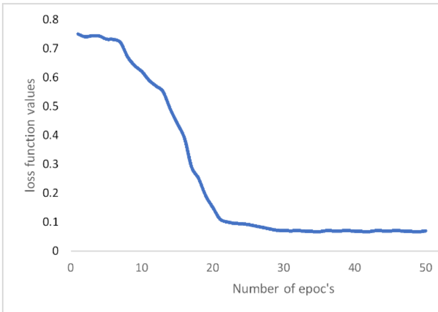
(c). Loss values on Comedy



(d). Loss values on Documentary



(e). Loss values on Romance



(f). Loss values on Horror

Figure 3. Training of the model on genre type Drama- Thriller- Comedy- Documentary- Romance- Horror

4. Ranking phase

Ranking phase starts after successful completion of classification phase. Our Ranking algorithm ranks the products considering following parameters. They are similarity of the image, average rating of the genre type and the user rating on that genre type. To find the similarity of the product images, algorithm uses last layers in the Classification Phase. The feature set of the last layers used to find the similarity of the images. Our proposed algorithm uses pairwise ranking approach. It calculates the scores of each possible pair of products in the dataset. Once the scores are calculated the products are arranged in sorted order based on their scores. Our algorithm provides the top-N recommendations when the user interact with the products. Precision and Recall metrics are used to evaluate our model. Our model achieved desired results.

Algorithm: Ranking Products

Input:

Dataset D= $\{p_1, p_2, \ldots, p_N\}$, the dataset (D) size is N, List T is list of top-N genre types from Classification phase,

M1[] [] # matrix M1 holds the pairwise similarity between products,

$avg_u$is users' average rating on the genre type,

avg_g is average rating on the genre type.

create () - Creates a matrix with NxN and fills with scores

Sort ()- Sorts the matrix based on scores

Tp-target Product

similarity ()- gets the similarity of the images.

Recommend () – Recommends diversity of products based on the genre types generated in Classification phase.

Step 1: Computing Similarity Matrix

Set i=0

while i< N do

whilej< N do

M1[][]= similarity(T,K)

j=j+1

i=i+1

end

Step 2: Computing Scores

Set i=0

while i< N do

for (j in d)

if (user given rating)

$avg_{u=} \sum$ j/N # average rating of user

score=(M1[][] * avg_u)

else

$avg_{g=} \sum$ j/N # average rating of genre type

score=(M1[][] * avg_g)

create(l1)

i=i+1

end

Step 3: Sorting the products based on scores

Sort(l1)

Step 4: Recommends products

Recommend(Tp, l1)

The above algorithm functioned with the foursteps. In the first step we calculate the similarity of each product with other product. This step carried out only once. Algorithm runs from step two to step four every time whenever a user needs recommendation of products. In the second step, we calculate the scores of each product. In third step, it sorts the products and keeps the highest scores acquired products in the top of the list. In the fourth step product recommendation takes places based on the genre type generated in the classification phase.

4.1. Result analysis

The performance of the proposed approach, for generating effective top-N genre products recommendations are discussed here. Our approach uses pairwise ranking approach for ranking the products. The proposed algorithm uses The Movie dataset which comprises of 6000 records. Our approach generated convincing precision and recall values. In our Ranking phase, precision is the ratio of intersection of relevant products and retrieved products to retrieved products. Recall is the ratio of intersection of relevant products and retrieved products to relevant products. precision@5 and recall@5 indicates that the top 5 products recommended which are similar and precision@10 and recall@10 indicates that the top 10 products recommended to the user which are similar.

Table 2. Performance of the classification model with ResNet-50 (50 Epochs) & ranked products metrics

Genre Type	Classification Metrics			Ranking Metrics			
	Accuracy	*Precision*	*Recall*	*Precision @5*	*Precision @10*	*Recall @5*	*Recall @10*
Drama	0.932	0.934	0.934	0.61	0.81	0.82	0.82
Thriller	0.908	0.902	0.902	0.58	0.78	0.8	0.81
Comedy	0.887	0.881	0.880	0.44	0.66	0.68	0.68
Documentary	0.864	0.876	0.877	0.36	0.59	0.63	0.65
Romance	0.835	0.836	0.837	0.32	0.57	0.59	0.6
Horror	0.804	0.805	0.806	0.3	0.55	0.58	0.58

5. Conclusion

This paper discusses product recommendation using deep Convolution Neural Networks. The paper is classified into 2 phases. In the first phase (classification phase), the products are classified using Convolutional Neural Network. In this phase, we used pretrained Convolution Neural Network architecture (ResNet-50) to classify the product images. In the second phase (ranking phase) the ranked products are recommended to the user. The performance of the work is discussed in result analysis for both the phases. We achieved good results. In future, we want to inculcate session-based information of the user to the system which would help in improving results.

References

Baldi P. (2012). Autoencoders, unsupervised learning, and deep architectures. *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, pp. 37-49.

Breese J. S., Heckerman D., Kadie C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. *Proc. of the 14th Annual Conf. on Uncertainty in Artificial Intelligence*, pp. 43-52. https://doi.org/10.1109/ICCV.2009.5459155

Cheng Z., Hurley N. (2009). Effective diverse and obfuscated attacks on model-based recommender systems. *RecSys '09: Proceedings of the Third ACM Conference on Recommender Systems*, pp. 141-148. https://doi.org/10.1145/1639714.1639739

Georgiev K., Nakov P. (2013). A non-IID framework for collaborative filtering with restricted Boltzmann machines. *ICML*, Vol. 28, pp. 1148-1156.

He K. M., Zhang X. Y., Ren S. Q., Sun J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778. https://doi.org/10.1109/CVPR.2016.90

He X. N., Liao L. Z., Zhang H. W., Nie L. Q., Hu X., Chua T. S. (2017). Neural collaborative filtering. *WWW Companion*, pp. 173-182. https://doi.org/10.1145/3038912.3052569

Hummel H. G. K., Van den Berg B., Berlanga A. J., Drachsler H., Janssen J., Nadolski R. J., Koper E. J. R. (2007). Combining social- and information-based approaches for personalised recommendation on sequencing learning activities. *International Journal of Learning Technology*, Vol. 3, No. 2, pp. 152-168. https://doi.org/10.2141/ijlt.10041

Krizhevsky A., Sutskever I., Hinton G. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, Vol. 25, No. 2, pp. 1097-1105. https://doi.org/10.1145/3065386

Lecun Y., Bottou L., Bengio Y., Haffner P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, Vol. 86, No. 11, pp. 2278-2324. https://doi.org/10.1109/5.726791

Li H. (2014). Learning to rank for information retrieval and natural language processing. *Synthesis Lectures on Human Language Technologies*, Vol. 7, No. 3, pp. 1-121. https://doi.org/10.2200/S00607ED2V01Y201410HLT026

Meteren R. V., Someren M. V. (2000). Using content-based filtering for recommendation.

Proceedings of the Machine Learning in the New Information Age: MLnet/ECML2000 Workshop, pp. 47-56.

Montaleão Brum Alves R. (2017). Information retrieval dataset - internet movie database (IMDB). *via Mendeley Data*. https://doi.org/10.17632/rth2kr5hxf.1.

Schafer J. B., Frankowski D., Herlocker J., Sen S. (2007). Collaborative filtering recommender systems. *The Adaptive Web*, pp. 291-324. https://doi.org/10.1007/978-3-540-72079-9_9.

Sedhain S., Menon A. K., Sanner S., Xie L. X. (2015). Autorec: Autoencoders meet collaborative filtering. *WWW Companion*, Vol. 3, pp. 111–112. https://doi.org/10.1145/2740908.2742726

Simonyan K., Zisserman A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv*, pp. 1409.1556. https://doi.org/10.1016/j.infsof.2008.09.005

Szegedy C., Liu W., Jia Y. Q., Sermanet P., Reed S., Anguelov D., Erhan D., Vanhoucke V., Rabinovich A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-9. https://doi.org/10.1109/CVPR.2015.7298594

Zeiler M. D., Fergus R. (2014). Visualizing and understanding convolutional networks. *European Conference on Computer Vision*, pp. 818-833. https://doi.org/10.1007/978-3-319-10590-1_53