JAISCR, 2023, Vol. 13, No. 2, pp. 63 – 72
10.2478/jaiscr-2023-0007

# A NEW APPROACH TO IMAGE-BASED RECOMMENDER SYSTEMS WITH THE APPLICATION OF HEATMAPS MAPS

Piotr Woldan[1], Piotr Duda[2,*], Andrzej Cader[3], Ivan Laktionov[4]

[1]*Department of Computer Engineering, Częstochowa University of Technology, al. Armii Krajowej 36, 42-200 Częstochowa, Poland Evorain Ltd. (www.evorain.com)*

[2]*Department of Computer Engineering, Częstochowa University of Technology, al. Armii Krajowej 36, 42-200 Częstochowa, Poland*

[3]*Information Technology Institute University of Social Sciences, 90 - 113 Łódź, Poland*

[4]*Department of Software of Computer Systems Dnipro University of Technology, 49005 Dnipro, Ukraine*

*E-mail: piotr.duda@pcz.pl*

*Submitted: 4th September 2022; Accepted: 15th February 2023*

**Abstract**

One of the fundamental issues of modern society is access to interesting and useful content. As the amount of available content increases, this task becomes more and more challenging. Our needs are not always formulated in words; sometimes we have to use complex data types like images. In this paper, we consider the three approaches to creating recommender systems based on image data. The proposed systems are evaluated on a real-world dataset. Two case studies are presented. The first one presents the case of an item with many similar objects in a database, and the second one with only a few similar items.

**Keywords:** feature extraction, recommender systems, heatmaps

## 1 Introduction

Nowadays, each company wants to attract as many customers as possible. The way to achieve the desired effect is to provide their customers with what they want. As a result, their portfolio includes countless numbers of products. It is a common issue in many branches like video streaming, online stores, or search engines, to mention a few of them. The tech giants like Google, Facebook, and Netflix are spending a wealth to create newer and better recommender systems [3]. The purpose of those systems is to provide a personalized list of products that can especially interest a given user. Various approaches have been considered to address the problem.

One of the most popular is collaborative filtering [14]. Its general idea is to find (in the entire database) the most similar users to the considered one. And the next step is to propose a set of products that are highly rated by this group but had not been considered by the chosen user yet. Besides the fact that this approach has been found beneficial in many cases, it still suffers from some drawbacks. As one of the major ones, we can point out that it cannot incorporate the user's taste. Although each person has specific interests, the provided recommendation will be more tailored to some group of people than to the given user. The other approach, the content-based recommendations, is based on a comparison of the products themselves [1]. The system extracts some features from each of them, and in the next step, it tries to find the most appropriate products based on the features' similarities. Products that are most similar to those that have been highly rated by the user are recommended. While this method allows adjusting the recommendation to the ratings of a specific user, it still does not allow taking into account their mood at a given moment. After all, it is not that the user always has uniform expectations. His previous ratings are the product of many different needs occurring at different periods. The study is currently underway to create a recommendation system that uses the user's tips on what they are looking for at the moment. If the user can clearly define his needs then he can use the knowledge-based approach. However, in most modern systems, it is not possible to predefine all user needs, and the users themselves are not fully aware of what they expect at the moment. To meet these needs interpretable recommendation systems are dynamically developing [12]. On the one hand, they allow the user to understand why a given item was recommended, but on the other hand, they can also help define the user's needs. This issue is especially difficult when it comes to the analysis of complex data, such as images [7], [6], [10], [11], [16]. Given the recommendations based on the pictures, it is not always clear what part of the image takes our special attention.

Based on the aforementioned motivation, we have decided to study the application of heatmaps to image-based recommended systems. In particular, the contribution of the paper can be summarized in the following points

- The usefulness of heatmaps to imaged-based recommender systems was examined.
- The new approch to create a feature map was propose. The proposition is based on the Grad-Cam algorithm.
- The experimental results was performed on *H&M* datasets

The rest of the paper is organized as follows. In section 2, the current studies on content-based recommender systems are presented. Section 3 describes all the approaches considered in this paper. The experimental results are presented in section 4, and the conclusions are given in section 5.

## 2 Related works

IIn recent years, the importance of recommendation systems has increased. Many articles showing the different ways of creating such systems have followed suit. In this section, we will focus on the latest papers about the content-based recommendation approach.

The article [5] examined the possibility of using CNN to create a clothing recommendation system. For this purpose, two independent networks have been created. Both networks are classifiers where the inputs were images of clothes. The labels for the first network correspond to the categories related to the target group of customers (men, women, children). The labels for the second network are categories related to the type of clothing. To create a single descriptor for a single image, the values from the last (12th) layer of both networks were concatenated. Consequently, the similarities of the images are based on the cosine similarity measure of 136-dimensional vectors. The topic of clothing recommendations was developed in [9]. The authors consider additional factors such as gender, body height, and clothing features. To gain information about body height, they applied a face detection mechanism. Moreover, the system consisted of two neural networks: CNN for gender recognition and GoogleNet for clothing attributes recognition. Next, they incorporate information about the texture of the clothing. The authors of [18] proposed a new hybrid method for image clustering using the combination of Particle Swarm Optimization (PSO) with k-means clustering algorithms. The image's features are based on colour histogram, colour moment, co-occurrence matrices, and wavelet moment. The paper is designed for content-based image retrieval, but it has obvious applications in recommender systems.

An approach that incorporated hand-crafted image features was presented in [2]. The authors obtained descriptors by application of SIFT feature extraction algorithm. They use the Bacteria Foraging Optimization algorithm to reduce the complexity, cost, energy, and time consumption. The similarities are obtained by the application of a deep neural network. Unfortunately, the details of this network have not been introduced. The paper [4] proposed the movie recommender system based on hybrid features. The hybrid nature of the features is related to their origin. The authors distinguished editorial, user-generated, and image-visual features. The visual characteristics features were determined by calculating the proportion of the different colour pixels for the whole image and then scaling them into one of eight categories.

Recently, several surveys presenting the current state of knowledge about recommendation systems have also been created. The graph-based approach is present in [8] and [17], and session-based approach in [15].

# 3 Features of the items

In this section, we recall the method which allows us to indicate key fragments of images. They are the crucial part that allows the interpretability of recommendations. We will be focused on two commonly used methods, i.e., guided backpropagation and unconvolution, and our modification of those methods. The methods allow us to create a feature vector to feather uses in the recommendation system. In the last subsection, we demonstrate the algorithm to create a recommendation system.

## 3.1 Various methods to extract regions of interest from an image

To visualize the most important image regions that have the most impact for its final assessment, the method of inverting the data flow through the network should be used. First, a high-level layer is established from which the signal is sent to the input of the network. As a result, we get a heatmap that has non-zero values only in selected areas. Various approaches can be used to reverse the data flow. The two most popular are backpropagation and deconvolution. Let us denote by $f^l$ the feature map established by the l-th layer of the convolutional network. For backpropagation, we need to know the gradient from the network output to the selected layer; let us denote it $R^{l+1}$. The gradient can be computed only if differentiable activation functions are applied in neurons. As ReLU is one of the most popular activation functions and is not differentiable on its whole domain then some modification should be used. The process can be represented by the following equation

$$R^l = (f^l > 0) \cdot R^{l+1}. \quad (1)$$

The above formula allows us to calculate the image representation for the lower layers of the network. Besides the issue of non-linearity of the ReLu function, transmitting gradients corresponding only to a positive value in the feature map produces good results. In contrast, the deconvnet method allows only positive gradients to go to lower layers, regardless of the value of the feature map itself. It can be expressed as the following equation

$$R^l = (R^{l+1} > 0) \cdot R^{l+1}, \quad (2)$$

or equivalently in a form

$$R^l = max(R^{l+1}, 0), \quad (3)$$

where the maximum values are computed for every component of the gradient $R^{l+1}$ separately. Finally, the authors of [13] propose a new method, called 'guided backpropagation', which combines both approaches. The method is given by the following formula

$$R^l = (f^l > 0) \cdot (R^{l+1} > 0) \cdot R^{l+1}. \quad (4)$$

Based on these methods, we want to propose a new one to improve heatmap extractions. The method proposed in (4) transmits only positive gradients; however, in a high-level feature, negative values play also a crucial role in a decision process so we decided to incorporate those regions into consideration. The proposed procedure can be expressed in the following form

$$R^l = (f^l > 0) \cdot abs(R^{l+1}), \quad (5)$$

where $abs(\cdot)$ is a function returning a feature map of absolute values of a given feature map.

### 3.2 Feature extraction

We will consider three approaches to extract features for the following use in the recommender system. The approaches are based on heatmap extraction methods described in subsection 3.1. In particular, we will use the deconvolution method given by (3), the guided backpropagation given by (4), and the newly proposed method given by (5).

Let us assume that the last convolutional layer ($l$) is composed of $k$ filters, and the output of this layer ($f^l$) is a tensor of shape equal to $(k, w, h)$. We have also computed the gradient from the output to the $l$-th layer, $R^{l+1}$. Now, we can define the three approaches md1, md2, and md3, given by

- **md1** $=> flatten\left((f^l > 0) \cdot abs(R^{l+1})\right)$,
- **md2** $=> flatten\left(max(R^{l+1}, 0)\right)$,
- **md3** $=> flatten\left((f^l > 0) \cdot (R^{l+1} > 0) \cdot R^{l+1}\right)$,

where $flatten$ is an operation that changes the size of the input tensor into the vector of size $k \cdot h \cdot w$.

### 3.3 Recommender system

The objective of our research is to analyze the performance of the recommendation systems based on product features, i.e., images. Therefore, we will focus on the content-based approach. One of its main advantages is its ability to deal with the so-called cold-start problem (a situation where we do not yet have user ratings for a given product). Such a situation is common, e.g., in the case of introducing new items to the offer, which is particularly common in the case of clothing sellers. Therefore, we will focus on creating a system that issues recommendations solely based on comparing product features. The process of indicating recommendations for a selected image will take place in the following steps:

- The first step is to create a neural network to extract features from an image. For this purpose, we can use a classification neural network. We can train the network from scratch or use one of the popular pre-train models like VGG, ResNet, or GoogleNet.
- Next, we have to extract the features. In our experiments, we consider three approaches, described in subsection 3.2. All of them are based on heatmaps described in subsection 3.1.
- The final step is to choose the most similar items. For this purpose, any similarity measure can be used. In our experiment, we will apply the cosine similarity measure.

The final step is to standardize the vector so that its values are integers between 0 and 255.

## 4 Experimental results

### 4.1 Dataset and system setup

To evaluate the performance of various approaches, we have used the H&M datasets. It is a real-world dataset based on the online store that proposes an extensive selection of products, mainly clothes. It contains the purchase history of customers across time. However, as we are focused on a specific task of giving recommendations based only on the image, as was described in section 3.3, we used only a set of images contained in this dataset. The original set consists of over 105k images divided into 19 groups (like Shoes, Garment, Full body, Garment lower body, etc.), where each group is divided into a number of types (like bootie, boots, heels). There are 130 types in total. To conduct the experiment we uses only part of this set consisting of 2244 images divided into 61 types of items. As shown in Figure 1, the number of elements belonging to each type is not uniform. The lowest numerous type has only one, while the most numerous type has 256 elements. The exact numbers of all types are given in table 1.

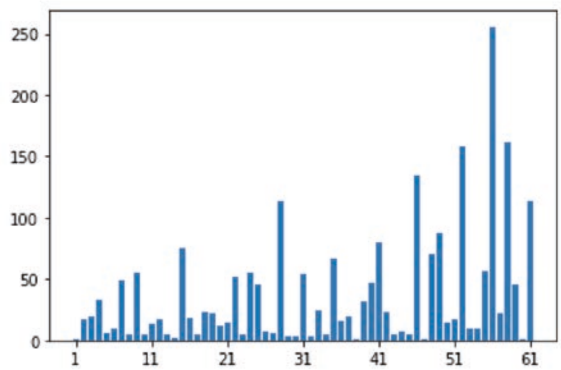


**Figure 1**. Number of elements of the types.

**Table 1**. Number of data elements in each type

| Type name | no. | Type name | no. | Type name | no. |
|---|---|---|---|---|---|
| Alice band | 1 | Bag | 17 | Ballerinas | 19 |
| Belt | 33 | Bikini top | 6 | Blazer | 10 |
| Bodysuit | 49 | Boots | 5 | Bra | 55 |
| Bracelet | 5 | Cap/peaked | 13 | Cardigan | 17 |
| Coat | 5 | Costumes | 2 | Dress | 75 |
| Earring | 18 | Felt hat | 5 | Gloves | 23 |
| Hair clip | 22 | Hair string | 12 | Hair/alice band | 15 |
| Hat/beanie | 52 | Hat/brim | 5 | Hoodie | 55 |
| Jacket | 46 | Jumpsuit/Playsuit | 7 | Kids Underwear top | 6 |
| Leggings/Tights | 113 | Necklace | 3 | Night gown | 3 |
| Other accessories | 54 | Other shoe | 4 | Polo shirt | 25 |
| Pyjama bottom | 5 | Pyjama jumpsuit/playsuit | 66 | Pyjama set | 16 |
| Robe | 20 | Sandals | 1 | Scarf | 32 |
| Shirt | 47 | Shorts | 80 | Skirt | 23 |
| Sleep Bag | 5 | Slippers | 7 | Sneakers | 5 |
| Socks | 134 | Straw hat | 1 | Sunglasses | 70 |
| Sweater | 87 | Swimsuit | 14 | Swimwear bottom | 17 |
| T-shirt | 158 | Tailored Waistcoat | 10 | Tie | 10 |
| Top | 57 | Trousers | 256 | Umbrella | 22 |
| Underwear Tights | 162 | Underwear bottom | 46 | Unknown | 1 |
| Vest top | 113 | | | | |

The VGG19 network was used as the classifier applied in the recommendation system described in section 3.3, see Figure 2. Nowadays, it is one of the most popular models for image classification. It maintains a good balance between accuracy and complexity. In addition, since the network consists of 19 layers, it works fast, which feature is important in the case of recommendation systems.

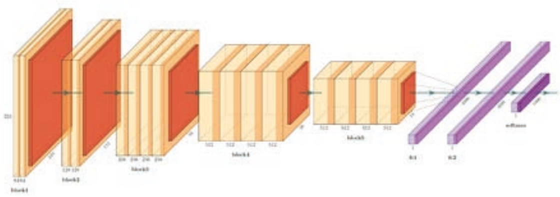

**Figure 2**. The schema of VGG19 network.

The numerical experiments were carried out on a computer with a 1.4 GHz quad-core Intel Core i5 processor, 8 GB of 2133 MHz LPDDR3 RAM and the Intel Iris Plus Graphics 645 1536 MB graphics card. The program was prepared in Python.

### 4.2 Case study

The effectiveness of the recommendation system is difficult to assess. There are no objective measures that would make it possible to unequivocally determine whether one system is better than the other in laboratory conditions. Only the execution of a given system and checking to what extent it helps users choose the content that interests them is reliable. The most effective method of comparing different systems is using the so-called A/B testing. In this approach, two recommendation systems operate within a given service. Users receive recommendations from a randomly selected system. The statistics of the decisions made on their basis allow the final evaluation of the models. In this work, the assessment is restricted to comparing the similarity of the obtained ratings. The results for each of the considered methods were presented on the basis of two case studies, an object taken form one of the most numerous types and from one of the least numerous types of products.

### 4.2.1 Lots of similar items

The most common clothes type in the considered dataset is trousers (256 items). Choosing a product from this group could turn out to be an easy task. Indicating other elements of this type as recommended products could result from the statistics of the dataset and not show the effectiveness of the methods considered. That is why we choose a product from the third most numerous type of clothes, t-shirts. This type has 158 elements. It is a number similar to other types such as Underwear tights (162 items) or Leggins (113 items). Finally, we chose image no 0260736033 for the study. We can see it in the top row in Figures 3, 4, 5, marked as a query image.

In Figures 3, 4, 5 we can see top-5 recommendation for a query image obtained by mp1-, mp2-, and mp3-based approaches, respectively. In the bottom lines, we see an image of recommended items with similarity measures above the image. As we can see, all considered methods provide similar recommendations; however, as recommended items belong to the same clothes type, t-shirt. We can see mutual items in pairs of recommendations, e.g., the brown t-shirt on the second position proposed by mp1- and mp2-based approaches, or the black t-shirt proposed by all approaches in positions first, third, and fifth, respectively. It is worth paying attention to the values of the assessment of product similarity. It seems that the mp1-based approach provides lower values than the other, and the mp3-based approach provides the highest ones.


**Figure 3**. The query image and five best recommendations proposed by the mp1-based approach.


**Figure 4**. The query image and five best recommendations proposed by the mp2-based approach.


**Figure 5**. The query image and five best recommendations proposed by the mp3-based approach.

Comparison of mean and median allow us to evaluate the variety of offered items. If the mean is higher, then it means that more items were scored similarly to the query image. If the median is higher, we know that only few items are similar, and the rest show higher differences. The standard deviation allows us to say whether the ratings are concentrated around the mean or are away from it. The computed statistics are presented in Table 2.

As we can see, mp1-based top 10 recommendation is the only case then the mean is lower than the median. Moreover, in this case, the standard deviation is relatively high. In the cases of mp2- and mp3-based approaches, the results are similar.

**Table 2**. The means, medians, and standard deviations obtained on the basis of the top 10 and top 40 recommendations by mp1-, mp2-, and mp3-based approaches. Case study I.

| **feature type** | 10 recommendations | | |
|---|---|---|---|
| | **mean** | **median** | **sd** |
| mp1 | 0.9689 | 0.9698 | 0.0076 |
| mp2 | 0.9718 | 0.9706 | 0.0035 |
| mp3 | 0.9748 | 0.9734 | 0.0052 |
| — | 40 recommendations | | |
| mp1 | 0.9539 | 0.9509 | 0.0103 |
| mp2 | 0.9595 | 0.9578 | 0.0087 |
| mp3 | 0.9601 | 0.9589 | 0.0103 |

Based on the case study, it seems justified that in the case of elements constituting a significant part of the considered dataset, the mp1-based approach can be especially useful to prose a few initial propositions. After that, the difference in ratings can importantly decrease. In the case of many recommendations, all the approaches provide similar results, and it is hard to indicate a single best method.

### 4.2.2 Few similar items

A skirt image was selected to test the performance of the recommender system for an item for which there are not many similar items in the considered set. This type contains only 22 elements. We decided that it makes no sense to carry out analyzes for the least numerous types because, in the case of types with only one or a few elements, the proposed recommendations must be purely random. As a consequence, an image with an index of 0356289005 has been selected. We can see it in the top row in Figures 6, 7, 8, marked as a query image.

In Figures 6, 7, 8 we can see top-5 recommendation for a query image obtained by methods mp1, mp2 and mp3, respectively. In the bottom lines, we see an image of recommended items with similarity measures above the image. As we can see, the proposed items differ significantly. The mp1-based approach proposes four skirts and one item of the other type. The other type of item is not the first recommendation. Since the difference between the considered type and the wrongly proposed one is obvious, it is worth noticing that the proposed item has a similar shape and colour as the query image. In the case of mp2-based recommendations, only two skirts were proposed among the best five. It should also be noted that the best match was chosen from different types of items. It is worth pointing out that similarities are higher concerning the mp1-based approach. Similar recommendations are carried out by the mp3-based methods. It also chose three items of the other type, and a skirt was not the first selection.


**Figure 6**. The query image and five best recommendations proposed by the mp1-based approach.


**Figure 7**. The query image and five best recommendations proposed by the mp2-based approach.


**Figure 8**. The query image and five best recommendations proposed by the mp3-based approach.

To demonstrate the variety of similarities assessed among the best recommendations, we compute the mean, median, and standard deviation based on top-10 and top-40 recommendations. The results are presented in Table 3. As we can see, the highest difference between the mean and median is in the case of mp1 based approach for top-10 recommendations (the mean is greater than the median). At the same time, this case has the highest standard deviations. It indicates that the initial recommendations are close to each other, and the later ones stick out. The lowest difference between mean and median and in value of standard deviation is in the case of mp2 based approach. It shows that all recommendations obtained similar results. In the case of 40-top recommendations, the mp3 provides the highest differences in similarity assessments, and the mp2-based approach again gives the most uniform recommendations.

**Table 3**. The means, medians, and standard deviations obtained on the basis of the top 10 and top 40 recommendations by mp1-, mp2-, and mp3-based approaches. Case study II.

| feature type | 10 recommendations | | |
|---|---|---|---|
| | **mean** | **median** | **sd** |
| mp1 | 0.9613 | 0.9574 | 0.0068 |
| mp2 | 0.9740 | 0.9738 | 0.0044 |
| mp3 | 0.9710 | 0.9688 | 0.0056 |
| — | 40 recommendations | | |
| mp1 | 0.9508 | 0.9492 | 0.0078 |
| mp2 | 0.9654 | 0.9635 | 0.0060 |
| mp3 | 0.9591 | 0.9562 | 0.0081 |

Based on the case study, it seems justified that in the case of elements constituting a small part of the considered data set and not radically different from other products (similarity between various clothes), we should choose a method based on the mp1 approach. The method is especially effective if we are interested only in a few recommendations. If we are interested in more of them, the mp3-based approach seems to be the better choice.

### 4.3 Iterpretability

To assess the interpretability of the considered methods, we examined the heatmaps used by each of the methods to create a feature vector. At first glance, all the approaches have a similar effect. It can be seen in the examples considered in subsections 1 and 2 (see Figures 9 and 10). The system focuses on the clothes' outline, applying different intensities to its various parts. The more intense the yellow colour is, the greater the values of the feature vector corresponding to a given area.

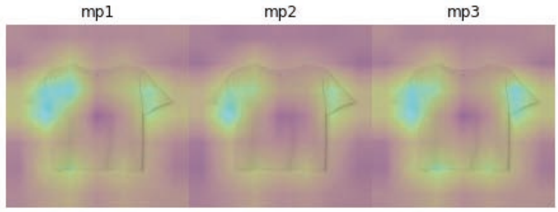


**Figure 9**. The heatmaps of image 0356289005 - mp1, mp2, and mp3 methods.

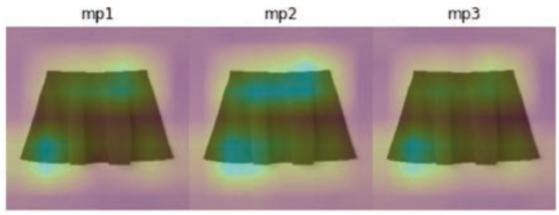


**Figure 10**. The heatmaps of image 0260736033 - mp1, mp2, and mp3 methods.

A closer analysis of the other images leads to the conclusion that the mp1-based approach provides the most intense feature vectors, the mp2 method is the least intensive, and mp3 is a balance between the two. We can see it in Figures 11 and 12. It does not change the fact that all three methods can be used to indicate the extent of interest.

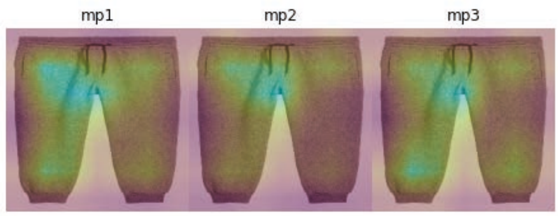


**Figure 11**. The heatmaps of image 0118458039 - mp1, mp2, and mp3 methods.

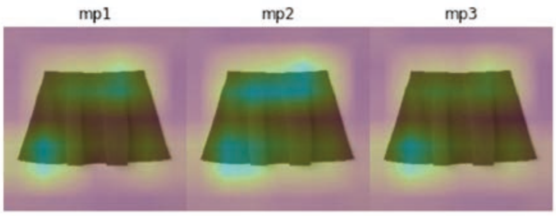


**Figure 12**. The heatmaps of image 0366686024 - mp1, mp2, and mp3 methods.

## 5 Conclusions

The approaches to creating recommendation systems studied in this paper proved to be effective. All of them ensured satisfactory results by providing recommendations in line with our expectations. The system using the proposed (mp1) approach turned out to be well suited to situations where there is a lack of similar elements in the dataset. Its benefits are particularly evident in the context of a small number of recommended items. The visualization with the application of the heatmap seems to be more intensive than in the other cases. Each of the approaches can be visualized. It is an additional advantage of the considered approaches. Proposed systems can be easily further explored by applying the methods to individual user preferences. It is an additional advantage of the considered approaches.

## Acknowledgments

This work was supported by the Polish National Science Centre under grant no. 2017/27/B/ST6/02852.

## References

[1] Charu C Aggarwal. Content-based recommender systems. In Recommender systems, pages 139–166. Springer, 2016.

[2] Aasia Ali and Sanjay Sharma. Content based image retrieval using feature extraction with machine learning. In 2017 international conference on intelligent computing and control systems (ICICCS), pages 1048–1053. IEEE, 2017.

[3] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. Wide & deep learning for recommender systems. In Proceedings of the 1st workshop on deep learning for recommender systems, pages 7–10, 2016.

[4] Fuhu Deng, Panlong Ren, Zhen Qin, Gu Huang, and Zhiguang Qin. Leveraging image visual features in content-based recommender system. Scientific Programming, 2018, 2018.

[5] Narges Yarahmadi Gharaei, Chitra Dadkhah, and Lorence Daryoush. Content-based clothing recommender system using deep neural network. In 2021 26th International Computer Conference, Computer Society of Iran (CSICC), pages 1–6. IEEE, 2021.

[6] Rafał Grycuk, Rafał Scherer, Alina Marchlewska, and Christian Napoli. Semantic hashing for fast solar magnetogram retrieval. Journal of Artificial Intelligence and Soft Computing Research, 12(4):299–306, 2022.

[7] Rafał Grycuk, Adam Wojciechowski, Wei Wei, and Agnieszka Siwocha. Detecting visual objects by edge crawling. Journal of Artificial Intelligence and Soft Computing Research, 10, 2020.

[8] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. A survey on knowledge graph-based recommender systems. IEEE Transactions on Knowledge and Data Engineering, 2020.

[9] Yun-Rou Lin, Wei-Hsiang Su, Chub-Hsien Lin, Bing-Fei Wu, Chang-Hong Lin, Hsin-Yeh Yang, and Ming-Yen Chen. Clothing recommendation system based on visual information analytics. In 2019 International Automatic Control Conference (CACS), pages 1–6. IEEE, 2019.

[10] Hung-Cuong Nguyen, Thi-Hao Nguyen, Jakub Nowak, Aleksander Byrski, Agnieszka Siwocha, and Van-Hung Le. Combined yolov5 and hrnet for high accuracy 2d keypoint and human pose estimation. Journal of Artificial Intelligence and Soft Computing Research, 12(4):281–298, 2022.

[11] Mirosław Pawlak, Gurmukh Singh Panesar, and Marcin Korytkowski. A novel method for invariant image reconstruction. Journal of Artificial Intelligence and Soft Computing Research, 11:69–80, 2021.

[12] Tomasz Rutkowski, Krystian Łapa, Maciej Jaworski, Radosław Nielek, and Danuta Rutkowska. On explainable flexible fuzzy recommender and its performance evaluation using the akaike information criterion. In International Conference on Neural Information Processing, pages 717–724. Springer, 2019.

[13] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving

for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806, 2014.

[14] Marko Tkalcic, Matevz Kunaver, Jurij Tasic, and Andrej Košir. Personality based user similarity measure for a collaborative recommender system. In Proceedings of the 5th Workshop on Emotion in Human-Computer Interaction-Real world challenges, pages 30–37, 2009.

[15] Shoujin Wang, Longbing Cao, Yan Wang, Quan Z Sheng, Mehmet A Orgun, and Defu Lian. A survey on session-based recommender systems. ACM Computing Surveys (CSUR), 54(7):1–38, 2021.

[16] Michał Wróbel, Janusz T. Starczewski, Justyna Fijałkowska, Agnieszka Siwocha, and Christian Napoli. Handwritten word recognition using fuzzy matching degrees. Journal of Artificial Intelligence and Soft Computing Research, 11(3):229–242, 2021.

[17] Shiwen Wu, Fei Sun, Wentao Zhang, and Bin Cui. Graph neural networks in recommender systems: a survey. arXiv preprint arXiv:2011.02260, 2020.

[18] Zeyad Safaa Younus, Dzulkifli Mohamad, Tanzila Saba, Mohammed Hazim Alkawaz, Amjad Rehman, Mznah Al-Rodhaan, and Abdullah Al-Dhelaan. Content-based image retrieval using pso and k-means clustering algorithm. Arabian Journal of Geosciences, 8(8):6211–6224, 2015.

**Piotr Woldan** graduated from the University of Technology in Częstochowa in 2015. He is currently a PhD student at the Częstochowa University of Technology, and the topic of his doctoral dissertation concerns hybrid structures as applied to recommendation systems. Since 2014, he has experience in research work related to business applications. He is a co-founder of Evorain sp.z o.o and passionate about algorithms, machine learning, and in particular deep learning methods, as well as parallel programming on heterogeneous architectures.
https://orcid.org/0000-0003-2067-2130

**Piotr Duda** received an M.Sc. degree in mathematics from the Department of Mathematics, Physics, and Chemistry, University of Silesia, Katowice, Poland, in 2009. He obtained his PhD and Sc.D. degrees in computer science from Częstochowa University of Technology, Częstochowa, Poland, in 2015 and 2019, respectively. His current research interests include deep learning and data stream mining.
https://orcid.org/0000-0001-7182-1349

**Andrzej Cader** is a professor at University of Social Science in Łódź, Poland. He received the Ph.D. degree in biocybernetics and biomedical engineering from the Medical University in Łódź, Poland. His research interests include intelligent systems science based on several architectures such as neural networks and complex systems. He is also currently engaged in time series analysis and chaos theory.
https://orcid.org/0000-0002-9107-5336

**Ivan S. Laktionov** is the Professor of the Department of Computer systems' software of Dnipro University of Technology (Dnipro, Ukraine). He received his Doctor of Science degree in 2021 majoring in computer systems and components. He specialises in advanced computer, sensor, infocommunication and microprocessor technologies of physical and chemical parameters monitoring and control systems. He is the author of more than 70 scientific works (paper, theses, monographs, tutorials and patents), including more than 20 articles in the world's authoritative publications Scopus and WoS.
https://orcid.org/0000-0001-7857-6382