DOI https://doi.org/10.35219/rce2067053276

Machine Learning in E-Commerce

Enache Maria Cristina
mpodoleanu@ugal.ro
Dunarea de Jos University of Galati, Romania

In this paper, methods and recommendations for an e-commerce web application of an online e-shop were explored, using various web technologies, such as Python. The approach used is described by analyzing the initial requirements, models and projects of the planned solution and the final implementation of the chosen method using a Model-View-Controller (MVC) framework. To achieve this goal, use case diagrams for specifications and implementation were developed, providing two levels of solutions: basic implementation using cookie functionality and advanced implementation based on the integration of machine learning algorithms. The adequacy and advantages / disadvantages of different methods, such as general recommendation systems and content-based recommendations, were analyzed and presented. This type of implementation is only a first step towards the explicable paradigm of artificial intelligence. Finally, the possibilities for future research are presented, taking into account several applications and other design aspects.

Keywords: e-commerce, machine learning, recommender system

1. Introduction

Customers are the center of all business. Every collective strategy, innovative technologies, intuitive business design try to improve the customer experience. The advent of artificial intelligence has loaded turbo approaches and helps define amazing business practices on a large scale. Moreover, by 2025, AI revenues are expected to exceed $ 30 billion. Personalized answers to customer questions, sending reminders about wish lists, and providing personalized alerts about shopping offers are some of them. Moreover, companies are dynamically inventing new ways of communicating to attract customers. AI contributes to the process with more accurate data and allows e-commerce stores to find the right customers. Likewise, how artificial intelligence (AI) has revolutionized industries is diversified, including in e- commerce, to check out the brilliant changes that AI has brought to the e-commerce industry and its benefits.

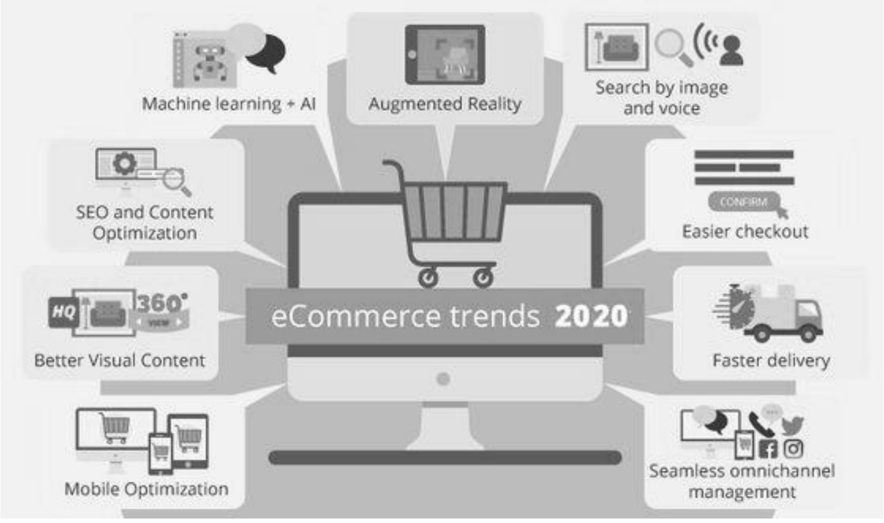

Fig. 1 - E-commerce trends in 2020

Technology brings significant changes to shopping habits. E-commerce improves the entire sale-purchase experience, both for sellers and buyers. Artificial intelligence for the user personalizes and improves the shopping experience. For the merchant, this is a means of improving sales and customer relations. This has become an integral part of how many consumers interact with digital platforms. The presence of artificial intelligence has led to close competition in e-commerce and has encouraged companies to use this technology to advance to the top of customer preferences. A deep understanding of artificial intelligence is an advantage for the e-commerce industry because it leads to a much better understanding of consumer behavior. Having a clear analysis of the buying behavior, one can design the perfect strategy with the company's objectives.

This paper is structured in two parts to build the problem of making adaptive applications of e-commerce using an increasingly used trend, namely machine learning. The paper aims to analyze the integration of web applications with adaptive features developed using framework web technologies by implementing an e-commerce case study. Two levels of adaptability are defined: basic, through the use of cookies and adaptability through the integration of machine learning algorithms. Adaptability can be defined as an interactive software system that improves its ability to interact with a user, based on partial interactions with that user. This improvement of the interface interaction can be achieved using information stored in the form of large data and processed by machine learning techniques, offering the user personalized recommendations. [6]

Machine learning algorithms can be implemented alongside a user interface using various web technologies. Recommendation systems are an example of the application of machine learning algorithms. Currently, these systems are used more widely when the application presents a low risk, such as recommending items in the store, due to their inexplicable nature and therefore lack of trust. However, the latest research on AI and its interfaces suggests that more is needed to build trust in AI decisions and interfaces that lead to the explainable AI paradigm. Explainable AI is a paradigm described by DARPA as a capacity that allows understanding, managing and essentially trusting AI, needed to solve the non-intuitive, opaque and incomprehensible nature of machine learning. With explicable AI, a chain of reasoning, based on knowledge and deduction of AI, can be provided to the user, demonstrating why the algorithm made certain decisions and not others.

Practical e-commerce applications that integrate artificial intelligence:

- Recommendation engines - Search engine artificial intelligence algorithms record details about the product you are looking for. Based on the results provided by the algorithms, the recommendation engine generates appropriate results and displays them on the user's dashboard to help him quickly find the right products.
- Chat bots - With the help of an artificial intelligence algorithm, computers are programmed to communicate with customers in a personalized way.
- Search for images - Buyers can search for products based on images. Mobile e-commerce applications can only find a product after it has been scanned using a camera. This process makes the classic keyword search useless
- Customer data management - E-commerce includes a variety of data (daily sales, stock variations, number of sales in a given area, customer data). It is impossible for a person to process all this data manually, so artificial intelligence is used. With its help, not only is the data collected much more structured, but a much better understanding of the results of the analysis is reached.
- Cyber security - Artificial intelligence has helped improve cybersecurity by detecting any fraudulent activity. To prevent cyber attacks and maintain a positive reputation for those who own an online business, artificial intelligence algorithms have been developed to detect any attempted fraud by hackers.

- Inventory management - An e-commerce platform has an extensive inventory of products, and artificial intelligence helps to manage it efficiently.
- After-sales services - Applications that use artificial intelligence can automate feedback forms and anything else related to products. By solving the buyer's problems, the value of the brand and the web site is improved.
- Customer Relationship Management (CRM) - Artificial intelligence can anticipate which customers are most likely to make a purchase, in order to communicate better with them. Artificial intelligence applications help identify future trends and can plan activities based on current trends. With the help of machine learning algorithms, customer relationship management (CRM) can be improved over time. [1]

2. Modelling and functionality

This paper starts from several functions of the administrator of an electronic store, such as:

- the function of viewing and searching for the products sold on the store's website
- the function of adding and deleting the products from the site
- the function of updating the details of a product, such as the stock level or the description
- link to a back-end database in which all product details must be stored.
- authentication system that authorizes two groups of users; general users and administrator users.
- displaying the list of products function in which the connected users will be able to leave reviews for individual products.
- review function, which allows connected users to view other users' reviews as well as add their own reviews.

In addition to the above functions that define the e-commerce system, components have been identified to implement a recommendation system. This took the form of a basic "Recommended Products" feature on the main page. Here, products related to those recently viewed by the user are displayed, anticipating that the user would also be interested in those products, with functionality similar to a content-based recommendation system.

For this paper, I chose to use Web2py, a Python web framework that uses the Model-View-Controller (MVC) paradigm. Web frameworks are increasingly used in web development due to the abstraction they provide for common and reusable web development tasks that allow the rapid development of applications with far fewer lines of code. Web2py is an open-source web application framework written in the Python programming language. Web2py allows web developers to program dynamic web content using Python. Web2py is designed to help reduce tedious web development tasks, such as developing web forms from scratch, although a web developer may build a form from scratch if required. [4]

Web2py was originally designed as a teaching tool with emphasis on ease of use and deployment. Therefore, it does not have any project-level configuration files. The design of web2py was inspired by the Ruby on Rails and Django frameworks. Like these frameworks, web2py focuses on rapid development, favors convention over configuration approach and follows a model–view–controller (MVC) architectural pattern.

In order to analyze and understand in a more efficient and complete way the system requirements, we will exemplify with the model of the use of the unified modeling language (UML). The use case diagram provides a description of how a user / actor of the system should perceive the whole system, ensuring that all the above requirements are met.

Figure 2 shows a usage case diagram consisting of several scenarios, the main actors being Users, Administrator and System and the actions of Registration, Login, Administrator

Login, Product View (user), Product Details Update (admin), Add / Remove Products (admin) and new product recommendation (system).

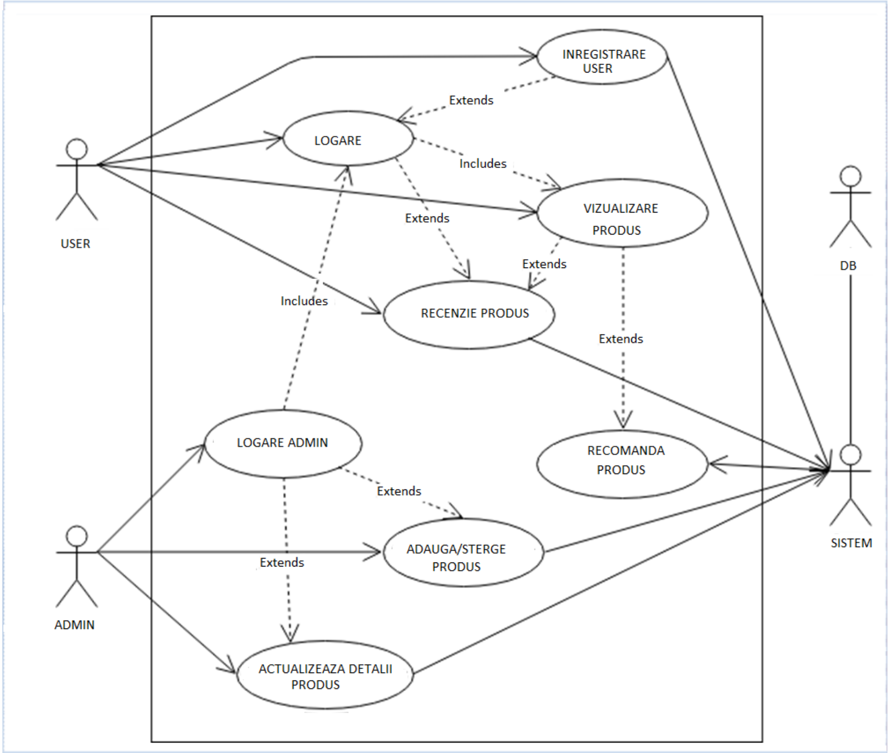

Fig. 2 –Use Case Diagram

In the case of product recommendation, the interesting feature of this diagram is the action of the system to recommend new products. In order to better capture the functionality of product recommendation, a use case was needed to understand the additional actions needed to identify items recommended to users. Fig. 3 presents a use case diagram that captures a more detailed approach to the actions required by the system to recommend new products to a user. This process is regardless of whether or not the user is logged in, as shown in Fig. 2.

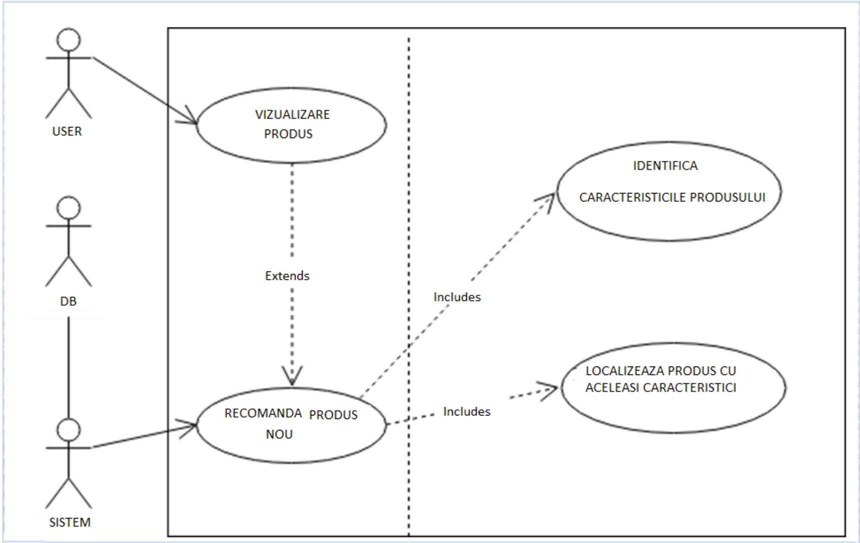
USER
VIZUALIZARE PRODUS
IDENTIFICA CARACTERISTICILE PRODUSULUI
DB
Extends
Includes
LOCALIZEAZA PRODUS CU ACELEASI CARACTERISTICI
RECOMANDA PRODUS NOU
Includes
SISTEM

Fig. 3. - Use Case Diagram for product recommendation

Starting from the use case diagrams detailed above, a system was initially developed using an MCV, which forms both the main e-commerce website and the product review application. The framework allows the administrator user to manage (add, edit, and remove) products, and regular users to view and write reviews of products that are visible to other users. In order to complete the functionality of the site with recommendation features, the main method is used in a first phase of cookies.

The code lines below shows that, first of all, cookies are created to contain information about the last product visited by a connected user. userID corresponds to the unique identification number for a user, and ultimProdus stores the identification number of the product that was last viewed by the logged in user.

```
def create ultimCookies(userID, ultimProdus): response.cookies[‘user_id’] = userID
response.cookies[‘user_id’][‘path’] = ‘/’ response.cookies[‘ultim_prod_id’] =
ultimProdus response.cookies[‘ultim_prod_id’][‘path’] = ‘/’
```

The code lines below shows the check performed to identify whether, when a user is logged in and has visited a product’s page, a cookie, containing the correct information, is set. If this check returns true, then the post variable is set to the last viewed product’s unique identification number. Then, possible items to suggest are identified using a characteristic, namely ‘categorie’, of the last viewed product. The function proceeds to select three items from the “produse” database, where their categories are the same as the category of the last viewed product. Using this logic, we are able to assume that, due to the user being interested in the initial product, they may also be interested in products from the same category which could be considered similar.

```
if request.cookies.has.key(‘user_id’) and request.cookies.has.key(‘ultim_prod_id’):
post =
db.produse(request.cookies[‘ultim_prod_id].value)
recomandare =
db.(db.produse.categorie == post.categorie)
.select(limitby=(0,3), orderby=~(db.produse.id))
```

Once the controller has selected appropriate items to recommend, presently stored in the “recomandare” variable, The code lines below shows the translation of these suggested products into the view for the user. Iterations of divider creation are completed for each suggested product, displaying product information such as name, and image. These products can now be viewed by the user.

```
<div class=”ProduseRecomandate”>
<div style=”width:60px;”>
<center>
<a href=”{{=URL(‘produse’, args =sugestie.id,1))}}”>
<p>{{=recomandare.nume}}</p>
<img class=”PozeProd thumbnail”
src=”{{=URL(‘download’,args=recomandare.image}}”/></a>
</center>
</div>
</div>
```

The recommendation of a product to the user is based on a product feature, which is in the product category. If the user has viewed a product in a category, it is assumed that they will like other products in the same category, therefore, several products in that category are recommended to the user.

Although this explanation is relatively simplistic, it demonstrates the limitations of providing recommendations using cookies. In order to give the user a more precise recommendation, several product features should be considered in addition to the product category. To this end, a data set containing each product and the definition of the characteristics should be constructed. For example, there is a popular application of book vase referral systems, so a book dataset would require information such as the name of the book and its description. To perform the content-based referral function, the data set will be used. To use a product's data, the Pandas library must read and manipulate the data. The Book data set will be used for this purpose.

Following this, an algorithm to identify similarities based on the item's description should be used. One such algorithm is Term Frequency-Inverse Document Frequency (TF-IDF) used to identify words or characteristics with strong relationships to the item they belong to [10].

TF-IDF stands for term frequency–inverse document frequency, a formula that measures how important a word is to a document in a collection of documents.

This metric calculates the number of times a word appears in a text (term frequency) and compares it with the inverse document frequency (how rare or common that word is in the entire data set). Multiplying these two quantities provides the TF-IDF score of a word in a document. The higher the score is, the more relevant the word is to the document. TD-IDF algorithms have several applications in machine learning. In fact, search engines use variations of TF-IDF algorithms to rank articles based on their relevance to a certain search query. When it comes to keyword extraction, this metric can help you identify the most relevant words in a document (the ones with the higher scores) and consider them as keywords. This can be particularly useful for tasks like tagging customer support tickets or analyzing customer feedback. In many of these cases, the words that appear more frequently in a group of documents are not necessarily the most relevant. Likewise, a word that appears in a single text but doesn't appear in the remaining documents may be very important to understand the content of that text. [9]

Suppose you are analyzing a review dataset: Words like this, if, this or that, will probably be among the most common. Then there will be a lot of content-related words with high frequency levels, such as communication, team, message, or product. However, these words will not provide too much detail about the content of each review. Thanks to the TF-IDF algorithm, you can evaluate the importance of each term and extract the keywords that best summarize each review. More specific words can also be extracted, such as multichannel, user interface or mobile application.

3. Conclusion

If you are active on social media or buy products from e-commerce sites, then you might have noticed that when you search for, or buy, products you start seeing ads for the same product or website. With the arrival of artificial intelligence, machine learning and big data play a big role in many industries. Some of those industries are healthcare, finance, automotive, government agencies, transportation, and retail. Companies use machine learning to help consumers make the right buying decisions.

Machine learning in e-Commerce helps companies make the most of their online store. E-commerce business owners can make better decisions, maximize sales, market their business effectively, and also understand their customers' requirements. As machine learning algorithms continue to learn to improve accuracy, there will be many more ways in which ML will benefit the e-commerce industry.

Recommendation engines are an inevitable variant of online stores. Amazon has revealed that about 35% of its revenue is generated by hiring a referral strategy. The recommendation of the relevant accessories and the good to have with the purchased product raises the cross-selling strategy. This further leads to an increase in sales and customer retention ratio.

The "customers who bought this item also bought" notification is rotated with these engines with appropriate machine learning techniques. By analyzing your previous shopping list and repeated page views, smart referral engines filter the result and help all e-commerce companies drive business. It looks like an experienced sales professional who can suggest the best products and accessories.

References

1. *Product data integration in B2B e-commerce - https://ieeexplore.ieee.org/abstract/document/941358*
2. *Supervised Machine Learning Applied to Link Prediction in Bipartite Social Networks - https://ieeexplore.ieee.org/abstract/document/5562752*
3. *https://monkeylearn.com/blog/what-is-tf-idf/*
4. *http://www.web2py.com/*
5. *A personalized products selection assistance based on e-commerce machine learning - https://ieeexplore.ieee.org/abstract/document/1382248*
6. *E-commerce Recommendation with Personalized Promotion - https://dl.acm.org/doi/abs/10.1145/2792838.2800178*
7. *SuperAgent: A Customer Service Chatbot for E-commerce Websiteshttps://www.aclweb.org/anthology/P17-4017.pdf*
8. *A graph model for E-commerce recommender systems*
9. *https://onlinelibrary.wiley.com/doi/full/10.1002/asi.10372*
10. *https://www.westagilelabs.com/blog/revolution-of-artificial-intelligence-in-e-commerce-in-2020/*
11. *https://monkeylearn.com/keyword-extraction*