UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSTITUTO DE CIÊNCIAS EXATAS
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Miguel Alvim de Almeida

ANALYSIS OF RECOMMENDATION METHODS FOR LEARNING OBJECTS USING LEARNING STYLES

Juiz de Fora

2020

UNIVERSIDADE FEDERAL DE JUIZ DE FORA

INSTITUTO DE CIÊNCIAS EXATAS

PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Miguel Alvim de Almeida

ANALYSIS OF RECOMMENDATION METHODS FOR LEARNING OBJECTS USING LEARNING STYLES

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Ciências Exatas da Universidade Federal de Juiz de Fora como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Orientador: Eduardo Barrére

Juiz de Fora

2020

Ficha catalográfica elaborada através do Modelo Latex do CDC da UFJF
com os dados fornecidos pelo(a) autor(a)

Alvim de Almeida, Miguel.
Analysis of Recommendation Methods for Learning Objects Using Learning Styles / Miguel Alvim de Almeida. – 2020.
51 f. : il.

Orientador: Eduardo Barrére
Dissertação (Mestrado Acadêmico) – Universidade Federal de Juiz de Fora, Instituto de Ciências Exatas. Programa de Pós-Graduação em Ciência da Computação, 2020.

1. Redes Neurais Artificiais. 2. Sistemas de Recomendação. 3. Estilos de Aprendizagem. 4. Felder & Silverman. 5. Similaridade de Vetor. 6. Objetos de Aprendizagem. 7. IEEE-LO. I. Barrére, Eduardo, orient. II.Título.

Universidade Federal de Juiz de Fora
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Programa de Pós-Graduação em
Ciência da Computação
www.pgcc.ufjf.br

Miguel Alvim de Almeida

"Analysis of Recommendation Methods for Learning Objects Using Learning Styles"

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do grau de Mestre em Ciência da Computação.

Aprovada em 03 de março de 2020.

BANCA EXAMINADORA

Prof. Dr. Eduardo Barrére – Orientador
Universidade Federal de Juiz de Fora

Prof. Dr. Jairo Francisco de Souza
Universidade Federal de Juiz de Fora

Carlos de Salles Soares Neto

Prof. Dr. Carlos de Salles Soares Neto
Universidade Federal do Maranhão

For our never ending hunger of knowledge. May it's source never ends, for life without the new, is but a vicious pointless cycle.

THANKS AND ACKNOWLEDGMENT

I would like to thank my family for helping me in all these years that I have dedicated to my academic studies, with special mentions to my Mother, Father and Sister.

I would also like to thank all my friends from the university which have helped and assisted me in my researches.

I also thank the Coordination for the Improvement of Higher Education Personnel (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES) of the Brazilian Government, whose financial help was of great help in keeping me focused on my research.

I could not forget to thank all the professors who taught me and assisted me to get this far, thank you very much.

Lastly I would like to thank Neusa Maria da Silva, who passed away before I had started studying for my bachelor of sciences degree back in 2012 and my late grandmother Nair. These women were role models of attitude and composure are part of the moral compass that I still follow to this very day.

To all the others who I could not fit here, I also extend my thanks.

The more I learn about learning and identifying what is learned, the more I realize that there is never a true end line to be reached, as there is always a new thread that links what was learned to another thing to be learned.

Perhaps that is the reason for our own existence as sentient beings, to always seek the end without ever reaching it, for if we ever reach it, life would be nothing but a meaningless vicious cycle.

I like to think that's how God keeps us going and maybe why we exist. We are ever hungry and never satiated.

So this is one of my contributions towards our never ending seeking of truth. I hope I made it a worth one.

On that same vein, we must also remember that if our objective is to never stop learning, then, as is put by Cousteau:

"Learning science, learning about nature, is more than the mere right of taxpayers; it is more than the mere responsibility of voters. It is the privilege of the human being."

Jacques-Yves Cousteau

RESUMO

Esta dissertação apresenta a pesquisa sobre a utilização de Estilos de Aprendizagem como critérios primários para a construção de um sistema de recomendação de Objetos de Aprendizagem. São identificadas as principais características necessárias para a criação de tal sistema, assim como quais passos são necessários para se utilizar Estilos de Aprendizagem durante um processo de recomendação. É proposta uma função utilidade para o processo de recomendação que faz uso de uma Rede Neural Artificial. Para a realização de testes, foram criadas bases de dados sintéticas de alunos baseadas no modelo de estilos de aprendizagem Felder & Silverman e de objetos de aprendizagem baseados no IEEE-LOM. Os resultados demonstram possíveis limitações inerentes do método de Anitha e Deisy para a classificação de objetos de aprendizagem para o modelo Felder & Silverman assim como da capacidade de utilização de estilos de aprendizagem como principais e únicos fatores para a recomendação de conteúdo educacional. Das conclusões tiradas, é visto que a adição de mais elementos pertinentes ao processo de educação e o maior estudo sobre a implementação e treino da rede neural podem ser o caminho necessário para suprimir e superar as limitações encontradas, que são o uso do método de Anitha e Deisy para converter objetos de aprendizagem do IEEE-LOM para o FSLSM e o uso exclusivo de estilos de aprendizagem como critério de recomendação.

Palavras chave: Redes Neurais Artificiais. Sistemas de Recomendação. Estilos de Aprendizagem. Similaridade de Vetor. Objetos de Aprendizagem. IEEE-LOM.

ABSTRACT

This master thesis presents a research about the utilization of Learning Styles as the primary criteria for the construction of a recommender a system for Learning Objects. The primary needed characteristics for the creation of such system are identified, along with which steps are necessary for the usage of Learning Styles during the recommendation process. It is proposed a utility function for the recommendation process that utilizes Artificial Neural Networks. So as to carry these tests, synthetic data databases containing students based on the Felder & Silverman model and learning objects on the IEEE-LOM model were created. The results show possible inherent limitations on the method of Anitha and Deisy for the classification of learning objects to the Felder & Silverman model and also to the capacity of learning styles to be used as the main and only factor for the recommendation of learning material. From the conclusions it is seen that the addition of more elements beyond the learning styles that are also pertinent to the learning process and a bigger study about the implementation and training of the neural network could lead to the necessary path to suppress and even overcome the aforementioned limitations, which are the usage of the method proposed by Anitha and Deisy for converting IEEE-LOM learning objects to the FSLSM and the exclusive usage of learning styles as criteria of recommendation.

Key Words: Artificial Neural Networks. Recommender Systems. Recommendation Systems. Learning Vector Similarity. Learning Objects. IEEE-LOM.

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF ABREVIATIONS

UFJF	Universidade Federal de Juiz de Fora
LS	Learning Style
FSLSM	Felder Silverman Learning Style Model
ILS	Index of Learning Styles
LO	Learning Object
LOM	Learning Object Metadata
ADALINE	Adaptive Linear Neuron / Adaptive Linear Element
MADALINE	Multiple ADALINE
ANN	Artificial Neural Network
RS	Recommender System(s) / Recommendation System(s)
IEEE	Institute of Electrical and Electronics Engineers

1 INTRODUCTION

In modern times most of our knowledge is stored into computerized databases in the form of digital media, such as text, audio, video, and images. This change from the traditional physical methods of knowledge storage brought up a much bigger ease of creating, storing and accessing content/information.

That said, the over abundance of information available brings up in a new issue: how to obtain useful information in this "ever-expanding sea" of documents/media? Seeking to tackle this problem, several studies into recovering relevant information of said databases have been and still are been done, in an area of research known as **Information Retrieval** (BAEZA-YATES; RIBEIRO-NETO, et al., 1999).

Information Retrieval works with the notion of finding documents that are relevant to a specific topic by analysing its contents and comparing them to key terms or ideas present the desired topic. It, however, does not goes into the merits of individual user preferences, in terms of identifying which, of said documents, are relevant for the particular needs and preferences of the user. To tackle that peculiarity, research into systems capable of profiling the needs/preferences of users and match them to the characteristics of stored documents and media was created and culminated into what is called **Recommendation Systems** (ADOMAVICIUS; TUZHILIN, 2005) or **Recommender Systems**.

Tailored search results created by Recommender Systems are the norm on the majority of modern commercial Information Retrieval services (such as the ones owned by Google, for example). A properly recommended material helps on the longer engagement of users with a system/platform, as they give to the users a better, more enjoyable experience.

Trying to bring some of these benefits to the realms of learning, this work seeks, through its research, to identify and collaborate with the possibility of automatically recommend educational material to students through the identification of their preferences profile and matching it to a list of relevant material stored on a specialized database.

1.1 MOTIVATION

There are several digital databases of educational material available both on the internet and on private servers, but few to none have recommender systems attached to their bases. Even the currently most popular video hosting service, Google's YouTube, which have a myriad of educational videos, has no specialized recommender system to cater to students particular needs and preferences (to their credit, they do have a recommender system, but its focus is to identify material that is related to what the user has already seen, not to what it needs).

Distance learning is becoming a solid alternative for those who are not capable to attend traditional learning environments, such as universities or colleges. However, it is also subject to much higher rates of student dropout (PIERRAKEAS et al., 2020). A method of working towards reducing such rates is to increase the engagement of the student with their learning activities by offering ones that are more appealing to him, something that a good working recommender system could definitely do.

1.2 OBJECTIVES

In order to study a possible method of profiling students and learning materials to be used on specialized educational recommender systems and identify its possible shortcomings and overall feasibility.

For such, two different methods were chosen (due to constraints in time and resources it was not feasible to do more methods):

- **Vector Similarity**: An already established and well known method.
- **Neural Networks**: A newer idea on the recommendation field (ADOMAVICIUS; KWON, 2015), with studies about its reliability for recommendation still being done.

1.3 CONTRIBUTIONS

As contributions of this master thesis, we intend to present:

- Create a case study over the reliability of using Learning Styles as the main factor of the process of learning material recommendation.
- An application of the method created by Anitha and Deisy (2015) to map objects defined by the IEEE-LOM (IEEE, 2002) metadata model to the Felder & Silverman learning style model, accompanied by an analysis of its benefits and possible shortcomings.
- A simple but properly documented artificial neural network recommender system.
- The creation of synthetic data sets of learning objects based on the IEE-LOM and of students based on the Felder & Silverman learning style model.

2 THEORETICAL DEFINITIONS AND FUNDAMENTALS

2.1 RECOMMENDATION/RECOMMENDER SYSTEMS

In large databases with a great number of stored materials, the capacity of identifying which ones are the most relevant for a specific user is not only a useful function but a needed one for the proper navigation and utilization of the base. This problem can be characterized as a situation where we have a database O with objects o such as $o \subset O$, a group of users U, with users u such as $u \subset U$ and the necessity to identify which $o \subset O$ are the most useful for each user in U.

As defined by Adomavicius and Tuzhilin (2005), recommender systems seek to treat this problem through the formulation of a utility function $Util$ that, for each object inside the database, is capable to ascertain its utility level for each possible user, allowing the recommender system to recommend objects that are more relevant for each user's needs. Therefore, it is possible to define a recommender system as a system that make use of an utility function $Util$ that returns to the user the object of highest relevance (rel):

$$\forall o \subset OS, rel = \max_{u \subset U} Util(u, o)$$

Adapted from (ADOMAVICIUS; TUZHILIN, 2005, p.3, equation. (1))

2.1.1 Types of Recommendation Systems

As presented by Adomavicius and Tuzhilin (2005), there are three ways for us to identify the utility of an object towards an user:

- **Content-Based Recommendation**: Recommends items similar to the ones the user preferred in the past.
- **Collaborative Recommendation**: Recommends items that people with similar tastes and preferences liked in the past.
- **Hybrid**: A mix of the two above.

2.1.2 Types of Criteria Approach

As shown above, there is more than one option to determine the utility of an object, since it needs to be taken into consideration what characteristics of said objects and users shall be used by the utility function. That generates the following classification:

Simple Recommender Systems use a **single criterium** to decide the relevance of an object. As a real world example, we have the way how some critics (be it for games, movies, books, and so on) quantify their opinion about a specific product, summarizing their piece of criticism as a single score which defines its overall quality.

This single score can be then utilized for ranking the many objects to be recommended. This method allows us to create very simple and straightforward utility functions.

Multi-criteria Recommender Systems use many criteria to measure the utility of an object. Following the critic example, we have a case where the criticism is summarized not as a single overall score, but in many key scores linked to specific points (in movies, for example, we can have, visuals, script, cinematography and so on).

This approach is more work intensive, as now our utility function must be able to identify what criteria are more relevant for each different user, with several cases where many are relevant, but with different intensities. This helps to create more precise systems, as the higher granularity allows for the better creation of a profile that better represents complex users and objects (ADOMAVICIUS; KWON, 2015).

Also, it is fair to note that,we can formally define the criteria relevance determination process spoken above with the use of the statistical term of **correlation** , which is usually measured as a value between 0% to 100% - [0,1] - used to shown how much two terms are related; there is no minimal or maximal amount of correlation necessary for us to classify a criterium as relevant; as long as the criterium helps the system to make an acceptable choice, it can be considered good.

2.1.3 Summarizing the Recommendation Process

We can summarize a method to create a recommender system as having the following steps:

- Definition and classification of a group of objects and users.
- Definition and classification of the characteristics of the objects and users to be used by the utility function.
  - Context of where the characteristics come from must be defined.
  - Characteristics properly quantified into numerical values with well defined domains.
- Defining a utility function that is compatible with all the chosen characteristics.

These steps are valid for both collaborative or content-based methods.

2.2 LEARNING OBJECTS

In general terms, learning objects (LO) are all and any materials and personal that are used and/or cited in the process of education and learning.

In a more formal manner we have:

> (...), a learning object is defined as any entity–digital or non-digital– that may be used for learning, education or training.
>
> (IEEE, 2002)

2.2.1 Learning Object Metadata - LOM

Given the abundance of digitalized LO, efforts to standardize any process of indexing culminated in what is known as metadata models of learning objects (Learning Object Metadata - LOM).

As the name implies, LOM are created with the intention of allowing the identification, quantification and indexing of characteristics that are pertinent and present in all learning objects to allow the indexing, search and recommendation of said objects, specially in the digital medium.

2.3 LEARNING STYLES

Learning Style refers to a theory that dictated that people learn in an easier way (faster and/or more efficient and/or engaging) in the cases where the material (learning objects and teacher didactics methods) that they have access to have certain characteristics that fit some of their personal preferences (COFFIELD et al., 2004).

In different words by Felder and Silverman (1988)

> A learning-style model classifies students according to where they fit on a number of scales pertaining to the ways they receive and process information.
>
> (FELDER; SILVERMAN, 1988)

This line of thought generated dozens of different theories that attempt to understand, identify and classify all the possible types of styles that people can have in learning (COFFIELD et al., 2004).

In general terms, learning style models are created with axes that are related to a cognitive preference of the student. Through some identification method - in general a multiple choice questionnaire (ALVIM DE ALMEIDA; BARRÉRE; SOUZA, 2019; COFFIELD et al., 2004) - the students' preferences are identified and quantified in practice creating a profile with all his characteristics.

With this profile each model gives a series of instructions or suggestions on how to better teach the students, with some models focusing on how to complement one learning and others dictating a full course, with specific classes and activities.

2.3.1 Criticisms and Flaws

Learning Styles also have their share of critics (COFFIELD et al., 2004; PHILLIPS, 2019; KIRSCHNER, 2017), who, correctly, point out several problems with the overall theory of learning styles and, mainly, with its implementations.

Many models of learning styles are built with the idea that they are capable of detecting and treating all possible peculiarities of each student, many times being seen as the main tool to be used for the selection of learning objects and class didactics (COFFIELD et al., 2004).

However, what is usually verified in practical applications is that learning styles do show tendencies, but said characteristics are not strong enough to, by themselves, allow us to properly define the best way to teach a student.

Therefore, a more proper application of learning styles is one that use than as a complementary information source in the process of teaching, using them to refine the already in use teaching methods, whose efficiency is already proved

2.4 VECTOR SIMILARITY

Vector similarity is a very simple geometric concept: The smaller the difference between each value in a vector has on the same value on other vector of same dimensions, the more alike both of them are.

In a very straight forward example, we have the vector A given by [0,0,0], vector B given by [1,0,0] and vector C given by [10,8,14]. It is very easy to see that vector A and vector B are much more alike than vector C, as the difference of the values stored on A and B is much smaller than the difference of values of A with C and B with C.

There are more than one way to measure said similarity, with the cosine value (cosine similarity) and the euclidean distance been commonly used.

2.4.1 Euclidean Distance

In this method the similarity is found by finding the distance between the defining points of both vectors by calculating the norm of the line created by these points. The smaller is the norm (with a minimum value possible of 0), the more alike the vectors are.

$$vector\ similarity = ||B - A|| = \sqrt{\sum_{i=1}^{n}((B_i - A_i)^2)}$$

2.4.2 Cosine Similarity

In this method the similarity of 2 vectors can be measure by analysing the cosine of the θ angle between them. The closer the value is to 1 (the cosine of 0^o), the more alike the vectors are.

$$cosine\ similarity = \frac{A \cdot B}{||A|| \times ||B||} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}}$$

This method has an advantage on the euclidean distance method when used on vectors of different intensities, were even if the magnitude of the vectors differs (and therefore the distance too), their angle will be the same. On cases were the vectors are shown with normalized coordinates, then both method should wield similar results.

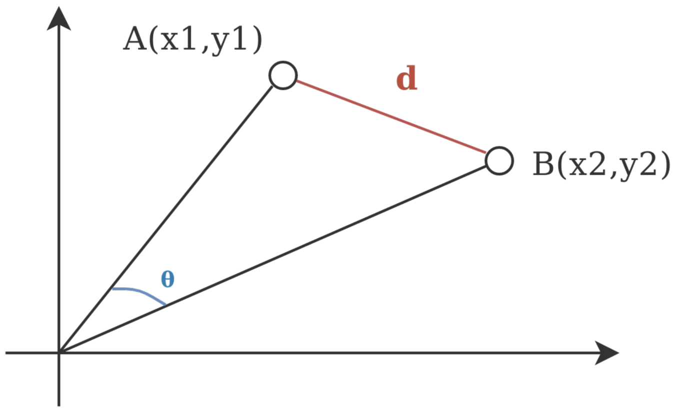

Fig. 1 – Example of Vector Similarity Calculation

Source: *https://cmry.github.io/notes/euclidean-v-cosine*

The difference between A and B can be found by calculating d or by finding the cosine of the θ angle.

2.5 ARTIFICIAL NEURAL NETWORKS

In the field of machine learning, artificial neural networks (ANN) are structures that seek to simulate, with a certain level of abstraction, the process of learning that the biological nervous system of animals use, by creating "entry neurodes[1]" that simulate the

1 in biology we have the neurons (the basic cells of the nervous system), whose links to other neurons, the synapses, are separated entities. In ANN we have neurodes, whose links to other neurodes are linked entities that belong to the neurode of origin. Therefore Neurons are biological terms and concepts and Neurodes are artificial terms and concepts.

many biological censorial systems present in real living beings. Then, these are capable of receiving numerical values (seen as signals) and then pass them along to the other neurodes of the network (organized into layers).

Each time a signal arrives into a neurode, it is firstly modified by adding a arbitrary number known as bias (usually a small number used to reinforce the signal and that can be modified during the network training process), then multiplied by a weight value that is linked to the connection of the neurode (its real world equivalent would be the synapses) that received and the one that sent the signal. Once multiplied by the weight, the signal is then transformed by an “activation function”, that is basically a mathematical function that modifies the signal value in a certain way. The transformed value is sent to all the other neurodes of the next layer until the final layer is reached.

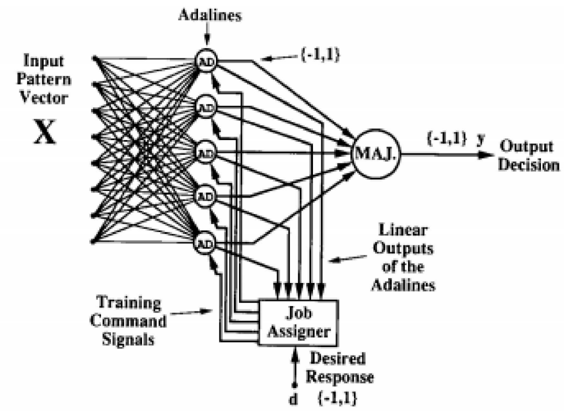

Fig. 15. A five-Adaline example of the Madaline I architecture.

Fig. 2 – Example of a Madaline Rule 1 Neural Network architecture

Source: Widrow and Lehr (1990, page 1426)
This network has 3 layers: a input layer, a hidden layer of Adaline neurodes and a output layer. Its training function is represented by the “Job Assigner”.

It is important to note that a neurode receives signals from all the neurodes from the layer before his and the signal that neurode transforms and passes forward is actually the sum of all the received signals multiplied by the weight of the respective link and with the neurode bias added to it.

In the final layer, known as the “output layer”, which usually composed by a single neurode, the received signal is transformed and the transformation value is considered the output of the network.

The aforementioned output is usually a binary or numerical value and is meant to represent the decision of the network about what the entry value (signals received by the entry neurodes) is.

Similarly to the other methods in the machine learning field, a neural network must be trained in order for it to obtain proper outputs.

In the more classic method of training, known as **supervised learning**, the learning comes from the process of comparing the output with the expected value of the entry. If a value is correct, nothing is done; if a value is wrong, the weights of the network links and the neurodes' biases are adjusted in order for the output to become correct. It is basically a learning by reinforcement where the correct is rewarded by keeping the network in it s current state and the wrong is punished by changing is structure so that it becomes correct.

The learning process may not be able to achieve 100% of precision, either due to the peculiarities of the type of data that the networks is trying to learn or due to the limitations of the training method used. In these cases the training is terminated once certain criteria chosen by the network creators is reached (maximum number of training iterations, reaching a threshold of precision and so on).

3 BIBLIOGRAPHIC REFERENTIAL

To properly understand the scenario with which we are working with, two separated researches into the known literature about learning styles, learning objects and recommender systems were done.

The first one focused on learning style models and learning objects current usage, and culminated in the research paper *"Learning Style identification and usage in academia: A systematic mapping"* (ALVIM DE ALMEIDA; BARRÉRE; SOUZA, 2019).

I this paper we read and analysed 102 different papers from a spam of approximately 20 years (early 2002 to mid 2019), utilizing databases such as Scopus, ACM and IEEExplore with the following search string:

(("Learning Style" OR "Learning Styles" OR "Estilo de Aprendizagem" OR "EstiLO de Aprendizagem") AND ("Learning objects" OR "learning object" OR "Objetos de aprendizagem" OR "Objeto de Aprendizagem"))

Utilizing the following criteria for inclusion and exclusion:

Inclusion Criteria:

1. The paper is in English or in Portuguese.
2. The paper discuss or presents the use of a Learning Style Model.

Exclusion Criteria:

1. The paper is not available.
2. No active use of a Learning Style Model (be it a specific one or a generic one - such as in the cases of general LO recommending systems) in conjunction with Learning Objects.
3. The paper is a systematic review or similar kind of work.
4. Usage of Learning Objects without a Learning Style (be it a specific model or a general idea of the application of a model).
5. The manuscript is not a full paper.

In the mapping we were able to identify the most used learning style to be the **Felder Silverman Model**, with more than 70% of usage on all the one hundred and two articles read. We also discovery a **small use of virtual learning environments**

in practical application researches along with an overall **low amount of attempts to justify the usage of Learning Styles** and **little to no discussion about their validity for the research done**. We were able to identified a few interesting works for our research and they we will be discussed more deeply bellow in their own subsection of this chapter.

The second one was carried out using the same digital databases from the first one, and was focused on understanding and identifying the main works on the field of recommendation. It was done through the use of 2 different search strings:

String 1:
((“Recommender” OR “Recommedation”) AND (“System" OR “Systems”))

String 2:
((“Recommender” OR “Recommedation”) AND (“System" OR “Systems”) AND (“Education”) OR (“Learning” (“Style” OR “Styles”)) OR “Learning Object” OR “Objects”)

In this particular case we did not created a full paper with our findings, as it was done in a more exploratory nature in order for us to properly understand the theory and paradigms of the Recommender Systems field. That said, we were able to find and study works that helped us in understanding and defining our own system. Also, after reading the work of Adomavicius and Kwon (2015) we were able to identify the potential of utilizing artificial neural networks as the core of our recommendation process. Following the proposition of utilizing artificial neural networks on our system we made a third search utilizing the following string:

String 3:
((“Recommender” OR “Recommedation”) AND (“System" OR “Systems”) AND (“Neural Network”) OR (“Neural Networks”))

Through that, we were able to verify an overall small amount of studies that focus on using neural networks in recommender systems, with one of the more completed ones being chosen to be explored bellow.

3.1 LEARNING AND TEACHING STYLES IN ENGINEERING EDUCATION

(FELDER; SILVERMAN, 1988)
Chosen as it is the article that proposed the Felder & Silverman Learning Style Model, which is the most, currently, used learning style model in academia currently, including on this master thesis work.

This article presents the conclusion from professors Felder and Silverman about student behavior and learning preferences that were observed during the many years where both taught for university undergrad students and was done by them perceiving that:

> Some instructors lecture, others demonstrate or discuss; some focus on principles and others on applications; some emphasize memory and others understanding. How much a given student learns in a class is governed in part by that student's native ability and prior preparation but also by the compatibility of his or her learning style and the instructor's teaching style. Mismatches exist between common learning styles of engineering students and traditional teaching styles of engineering professors. In consequence, students become bored and inattentive in class, do poorly on tests, get discouraged about the courses, the curriculum, and themselves, and in some cases change to other curricula or drop out of school
>
> (FELDER; SILVERMAN, 1988)

They focused on understanding and categorizing how engineering teachers' teaching methods conflict and differentiate from students' learning preferences. These observations culminated into the formulations of their own learning-style model.

The created model borrows dimensions and concepts from other works, such as Carl Jung's theory of psychological types (JUNG, 1971) and Kolb's own learning style model (KOLB, 1984).

Five axis containing 2 opposing dimensions each exists in the model, giving a possible total of $2^5 = 32$ possible distinct styles.

Although initially daunting, the article goes into details on how each dimension works and on how teachers may address them, showing a great amount of overlapping and how that helps in the process and concludes that their model and techniques should help teachers to address their student's problems of focus and engagement in the class.

It is important to note that later on the total amount of axis reduced to 4 and the dimensions to 8 with the removal of the *Inductive/Deductive* axis and dimensions; the current available version of the article has a preface of professor Felder explaining why. Also the *Visual/Auditory* axis was changed to *Visual/Verbal.* This particular article is the creator of the FSLSM, but does not get into the merits of verification and validation of

Preferred Learning Style		*Corresponding Teaching Style*	
sensory intuitive	perception	concrete abstract	content
visual auditory	input	visual verbal	presentation
inductive deductive	organization	inductive deductive	organization
active reflective	processing	active passive	student participation
sequential global	understanding	sequential global	perspective

Fig. 3 – Original 5 Axis and 10 Styles from the FSLSM

Font: (FELDER; SILVERMAN, 1988)

the same. That said, validation was later done on the model, as is indicated by Coffield et al. (2004).

3.2 LEARNING STYLES AND PEDAGOGY IN POST-16 LEARNING: A SYSTEMATIC AND CRITICAL REVIEW

(COFFIELD et al., 2004)

Chosen as it is one of the most influential works on the are of Learning Styles, presenting a proper definition of Learning Style Models along with a detailed and through evaluation of a dozen of the, at the time, most used learning style models.

This 2004 review still is used as a critical article to discuss and understand Learning Styles. In its 182 pages, a complete report about 12 learning styles models is conducted and more than 60 styles are shown and quickly analysed.

A thorough explanation about the basics behind the learning styles theory is presented, along with a taxonomy system to better understand and categorize each model. Named "families", they are presented in a list on the figure 4 bellow.

The research done in this article helped to mold the current scenario on learning styles, by aiding to create a vision on the minds of several researches and by pointing several critical points of failure and success on the many theories discussed, such as weak or questionable validation methods, overuse of learning styles on the education system, ill defined terms and it's effects and lack of proper scientific composure on the part of some authors.

Although still relevant to this day, this work shows some points of conflict with the modern paradigm of the learning styles applications and realities. As put by us in a different article:

> In [Coffield et al. 2004], a thoroughly systematic review of several Learning Styles Models and their classification questionnaires is done, getting into the merits of their validity, application and on which elements inherent of the person doing the questionnaire are used for their Learning Style classification. This work is cited in several other relevant works in the LS field and might be the most complete review on the subject to this date. **However, this review does not touch on some of the most currently used Learning Style Models, like the Felder & Silverman and VARK models, what does not help if ones seek to understand the current state of usage and validity of the field.**
>
> (ALVIM DE ALMEIDA; BARRÉRE; SOUZA, 2019)
> Bold text added by this dissertation.

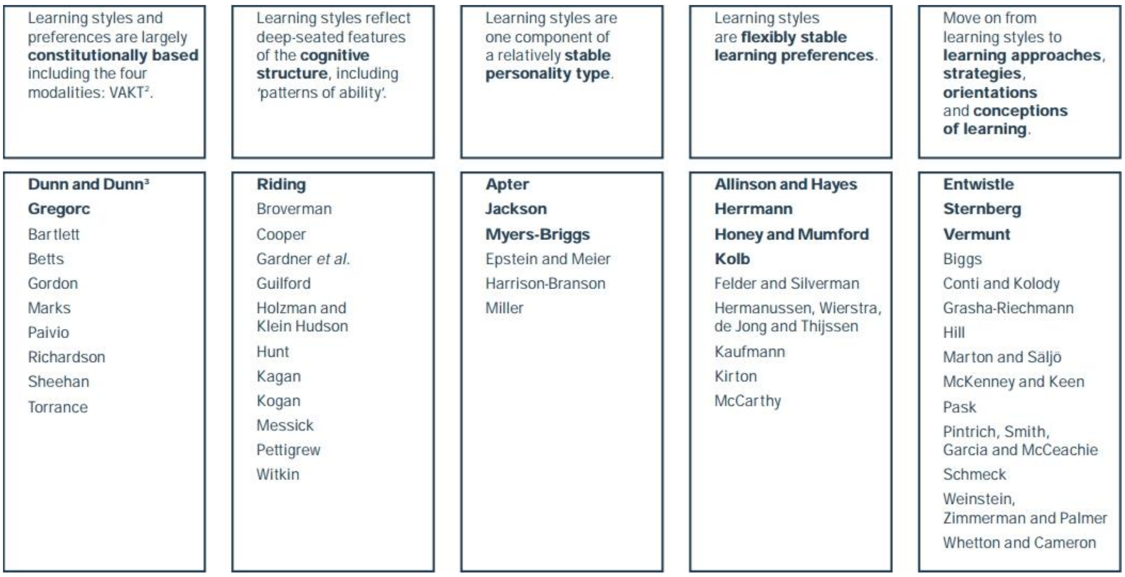

Fig. 4 – Coffield et. al 2004 Learning Style Families

Font: (COFFIELD et al., 2004, Page 19)
The names in bold are the models examined in detail on the revision.

3.3 STOP PROPAGATING THE LEARNING STYLES MYTH

(KIRSCHNER, 2017)
Chosen as it offers a counter balance to the idealist vision that some proponents of learning style model users have towards their usage in education. It is important to point the limitations of the tools used on any work (like learning styles)and address them.

In this open letter, professor Paul A. Kirschner touches on the merits of the entire Learning Style theory. With a very strongly oppositional stance, the letter points out a lack of evidence of validity on the entire concept of learning styles and how many models, as put by the author, “pigeon hole” students into set, non flexible and exclusive classifications that many times are ill suited for all the necessary purpose.

Is also pointed out a lack of strong reliability on the many tests done to classify a student learning style (a fact also pointed by Coffield et al. (2004)) and the low rate of correlation of self reporting preferences by students and actual proper methods used for learning, a case of what you want is not what you need.

The letter closes by pointing out that many “proofs” given by learning styles users is anecdotal at best and do not have the necessary rigor to be properly considered.

This letter brings strong and well based arguments against the use of learning styles, tackling mostly the usage of them as tools to shape and create entire courses and activities around. As pointed by Alvim de Almeida, Barrére, and Souza (2019), currently there are many efforts on using learning styles for less intrusive and critical activities, like in this work itself where they are used as a auxiliary tool to enhance a already established learning system. In the end, it is all about measuring expectations and taking what the tool can actually do into consideration.

3.4 MULTI-CRITERIA RECOMMENDER SYSTEMS

(ADOMAVICIUS; KWON, 2015)
Chosen as it is a extremely complete revision of the whole recommender systems field, offering the necessary basic definitions for work and, most importantly, a call towards the exploration of the use of neural networks.

In this extensive and comprehensive article, Gediminas and YoungOk explain and define all the major points, classifications and techniques used on recommender systems, getting into the merits of the theoretical and historical background of this research and application field.

The definitions used on this master thesis are based on this work and on a previous work by Gediminas as well (ADOMAVICIUS; TUZHILIN, 2005). This article is a great starting point for those seeking to understand the theory behind automatic computerized recommendation, and although it will present every new advanced personalization technique, it will also provide the necessary basics to properly start working.

Also it is important to note that this article is credited as being one of the works who proposed the utilisation of artificial neural networks on the recommendation process, making a call for research on the topic in it's closing statements.

3.5 PERFORMANCE ANALYSIS OF NEURAL NETWORKS-BASED MULTI-CRITERIA RECOMMENDER SYSTEMS

(HASSAN; HAMADA, 2017a)

This article was chosen because it offers a basis for utilizing artificial neural networks on recommender systems by showing it's capabilities to match and surpass more widely used methods.

Hassan and Hamada have done several works on the field of recommender systems. In this one in particular, seven metrics of measurement are used on a multi-criteria recommender system based on a neural network, with empirical results showing a high prediction rate and good recommendation results from the system proposed.

The evaluation metrics used where:

- RMSE - Root mean square error
- MAE - Mean average error
- Precision
- Recall
- Accuracy using F_1 measure
- FCP - Fraction of concordant pairs
- ROC/ROCAU - Area under the curve of the receiver-operating characteristic

This article explores the difficulties on modeling a multi-criteria recommender system and how to used it with a neural network trained with the delta rule technique. It explores the merits of using a neural network as the basis for the utility function of the recommender system, inspired by the work of Adomavicius and Kwon (2015).

The results are promising but it is pointed that many more questions need to be answered, such as: how many layers a artificial neural network (ANN) can use to increase the results? What training algorithm should be used? What are the best criteria for the characteristics of the users and objects? Would deep learning help?

4 DEVELOPMENT

4.1 DEFINING THE PROBLEM

As shown in the introductory chapter, this work's focus is to study recommendation methods of learning objects for students with the goal of defining the possibility of using them as tools to enhance their access to useful learning material. Therefore, in general terms, we can define our recommendation problem as a situation which there is a group of learning objects of which we wish to know the ones that are more compatible to be given to each student in any group of students that might come to use our learning object group.

In more formal terms: There is a group of students E and a group of learning objects OS and we want to obtain a list that contains all elements $o \in OS$ organized into a descending manner in terms of relevancy with any $e \in E$, which relevancy refers to how much said elements that belong to E and OS display characteristics that have good correlation between themselves.

That gives us a variation of the basic definition of the recommendation problem as given by Adomavicius and Tuzhilin (2005), where:

> In its most common formulation, the recommendation problem is reduced to the problem of estimating ratings for the items that have not been seen by a user.
>
> (ADOMAVICIUS; TUZHILIN, 2005, p. 2)

Still using the work of Adomavicius and Tuzhilin (2005), we can formulte our problem as the search for the function $Util$ that measures how much a $o \in OS$ is useful to a $e \in E$, giving us a recommender system that attempts to:

$$\forall o \in OS, rel = \max_{e \in E} Util(e, o)$$

4.2 IDENTIFYING THE ELEMENTS OF THE RECOMMENDATION PROCESS

Following what was defined into chapter 2 as necessary elements for a recommender system we have:

4.2.1 Defining the Group of Objects and Users

As mentioned earlier in the section “DEFINING THE PROBLEM”, the elements to be used in the recommendation process are:

- **Learning Objects**: The objects to be recommended.
- **Students**: the users that will receive the recommendation.

4.2.2 Defining the Characteristics of the Objects and Users

To generate a profile of our users, we chose to use the learning style model created by Felder and Silverman (1988). Known as FSLSM (*Felder and Silverman Learning Style Model*), this choice was made by the facts that the FSLSM:

- Is the most used learning style model in related researches (ALVIM DE ALMEIDA; BARRÉRE; SOUZA, 2019), what makes the process of finding working material in general easier.
- Has four axes, each with 2 values that are directly inversely proportional on each other. That means that the FSLSM has 8 characteristics for us to work with our recommending process.
  - Each axis has a total combined value of 100% that must be distributed between its two characteristics. That means that the FSLSM already gives us quantified characteristics in a close interval between 0 and 1, what is excellent for the process of machine learning and vector similarity we intent to use.
  - Each value in the FSLSM student profile can be defined as:

$$value \in \mathbb{R}$$

$$value \subset [0, 1]$$

- Its official style classification method, the Index of Learning Styles (ILS) (FELDER; SOLOMAN, 1997) is a simple, fast and free to use tool. The test measures each of the model's 4 axis into values ranging from -11 to +11 (the extremes just mean that one user is more attuned to one style and less attuned to the other style from the same axis; also the possible values for each axis are all odd numbers, as there are 11 questions per axis and their answers can only add 1 or -1 to the axis final score) and offers 3 levels of preference based on those numbers (Neutral, moderate preference and strong preference). These levels are useful parameters for our utility functions.
- Although criticized as unable to, by itself, point out the better methods of teaching a student, learning styles (and therefore, the FSLSM) can still be used as tools that help us to find out relevant characteristics in students that will allow us to create recommending systems without problems such as the "cold start", as it takes less than 10min to measure one's learning style. As we stated in a previous work (ALVIM DE ALMEIDA; BARRÉRE; SOUZA, 2019), with learning styles, is all about measuring expectations.

In order to generate a profile of our objects that is compatible with our users, we decide to use the metadata model IEEE-LOM (IEEE, 2002). The main reasons are:

- Just like the FSLSM, the IEEE-LOM is a model already in use on several other works (ALVIM DE ALMEIDA; BARRÉRE; SOUZA, 2019).
- The work of Anitha and Deisy (2015) proposes a method to convert soome characteristics of the IEEE-LOM into the 4 axis of the FSLSM. That, in theory, gives us a way to measure the correlation between our objects and users in a very straightforward manner. It is worth noting that by mapping the IEEE-LOM into the FSLSM we can also make use of the well defined and well behaved values of the FSLSM (their domain and so on), as explained above in this chapter.

4.2.3 Defining the Utility Function

To define this last aspect of our system is necessary to also define some minor characteristics:

- **System Type**: In the created system we opted for an approach that can be defined as **based on content**. However it has to be noted that the system does not use pass searches as the formal definition states, as we only make use of the learning style. The system uses an approach that uses content based on what the type of profile of the user was previously classified as having. Users with similar or identical profiles will have similar or identical results.
- **Multi-Criteria System**: As seen above, our method will use 8 characteristics for the recommendation process, what by definition, classifies us as a Multi-Criteria System.

As stated, our study uses two different functions, a **vector similarity based one** and a **neural network based one**.

4.2.3.1 Vector Similarity Utility Function

Since we are working with 8 dimensions (the 8 characteristics presented by the FSLSM), each student and learning object can be represented as a point in a 8 dimension plane, what allows us to create vectors of users and objects and compare them for similarity. In other terms: Each attribute of our utility function (user and object) can be seen as a coordinate of a vector that originates from the origin point and we can compare vectors to obtain a similarity value.

Since all of our values are already normalized into [0,1] thanks to the FSLSM classification, the choice of euclidean distance or cosine similarity fell down to personal preference and in the end we opted for the use of the euclidean distance as our similarity function:

$$similarity = \sqrt{\sum_{i=1}^{n}((B_i - A_i)^2)\alpha}$$

The α parameter was added to the formula to allow us to weight different characteristics of the FSLSM based on the users preferences by using the ILS classifications of affinity (as said above: Neutral, Moderate and Strong). Simply put, we set:

- Neutral Affinity (ILS values between [-3,3]): $\alpha = 0.5$ for both
- Moderate Affinity (ILS values between [-7,-5] or [5, 7]): $\alpha = 0.75$ for the moderate side and 0.25 to the weaker side
- Strong Affinity (ILS values between [-11,-9] or [9, 11]): $\alpha = 1$ for the strong side and 0 to the weaker side.

The reason for such a thing is due to the fact that if, in theory, an object has a strong score for the Reflexive characteristic and the user has a low affinity for this characteristic, then the euclidean distance will be significantly increased, as the difference between the values would be substantially big. This would result in a big penalty to the object's overall similarity score, even if all the other 7 characteristics actually were similar. That is not desirable as, per the FSLSM, this particular characteristic is not very relevant for the student and a high or low amount of it in the object will not matter for the student at all.

With this reasoning, we can use α to weight down characteristics that do not matter for each particular user and increase the possible accuracy of our function.

4.2.3.1.1 Process of Recommendation

A vector with all 8 characteristics for the user has its similarity tested with the utility function for all objects on a data base. The results are compiled and ordered in a vector in ascending order. With that done the user is capable of picking the N most recommended objects by getting the first N objects from the objects vector.

4.2.3.2 Neural Network Utility Function

We are not the first ones to attempt to use neural networks for the process of recommending something in multi-criteria recommender systems (HASSAN; HAMADA, 2016, 2017b,a). In the article *Performance Comparison of Feed-Forward Neural Networks Trained with Different Learning Algorithms for Recommender Systems* (HASSAN;

HAMADA, 2017a), a neural network was used in a recommender system and then compared to many other recommendation methods with good results. Based on those results, we built a network using the same basic structure used by the article, with the information available on it.

That lead us to create a MADALINE network that is composed of several ADALINE (WIDROW; LEHR, 1990) networks. A ADALINE network uses the *Delta Rule Algorithm (DRA)* (also known as the Widrow & Hoff Learning rule or the Least Mean Square (LMS) rule) for training.

> The DRA is a popular and efficient algorithm for training ANNs that does not contain hidden layers. It is based on a gradient descent algorithm that was developed to train two-layer networks which deal with a nonlinearly separable data set. It uses a constant learning rate η, which is a parameter that controls how much the updating steps can affect the current values of the weights. It also contains the derivative of the activation functions $f\omega(X)$, the error between real and estimated outputs, and the current features to compute $\Delta\omega$ (see (3), yj is the actual value from the data set) so that the updated weights that can be used in the $n + 1$ iteration will be computed using $\Delta\omega$ (see (4)). Note that in this experiment, the DRA algorithm may be referred to as ADAptive Linear Neuron (Adaline) algorithm.
>
> $$\delta\omega_i = \eta(y_i - f_\omega(X))f'_\omega(X_i)x_i$$
>
> $$\omega(n + 1) = \omega(n) + \delta\omega$$
>
> (HASSAN; HAMADA, 2017a)

As seen in the citation above, the DRA is the basic algorithm used on the ADALINE and MADALINE networks and is the one we opted to use as well. That said, the DRA is applicable in single layer networks, therefore, an adaptation is necessary to apply it to a multi-layer network like the MADALINE.

The works of Widrow and associates offers 3 different methods (WIDROW; LEHR, 1990) to implement the concepts of the DRA into a MADALINE design. The base article from Hassan and Hamada did not went into details on what method was used, therefore we opted for using the Method 1 (WIDROW, 1962) as all 3 methods are equally capable of training a network (WIDROW; LEHR, 1990) and it fell down to a choice in easiness of implementation.

Known as **Madaline Rule 1 (MR1)**, the algorithm is explained by Widrow and Lehr (1990) as:

> The MR1 rule allows the adaptation of a first layer of hard limited (signum) Adaline elements whose outputs provide inputs to a second layer, consisting of a single fixed-threshold-logic element

> which may be, for example, the OR gate, AND gate, of majority-vote-taker discussed previously. The weights of the Adaline are initially set to small random values.
>
> (WIDROW; LEHR, 1990, page 1426)

Figure 2 in the chapter 2 shows a example of a Madaline Rule 1 network.

In our implementation we made use of a OR gate for our output layer, with the following implementation:

This implementation of the MADALINE network is cable of answering yes (1) or no (-1) only, meaning that a single network can only detect if a object is suited in consideration to one single characteristic[1].

The ILS initially, as stated above, gave us 3 levels of affinity: weak when we have between 1 and 3, moderate with 5 to 7 and strong with 9 to 11. However those are not enough in this case, as the method given by Anitha and Deisy (2015) does not follow the ILS score balance and a single object can be, for example, strong on both elements of a axis and even vice-versa. Therefore, we are not capable of measuring the worthiness of both elements of an axis by only looking to a single score (the ILS score), we need to verify both.

To solve that issue we made modifications to the original 3 levels and added a new affinity level:

- **Not Relevant**: Less than 25%
- **Weak**: Less than 50%
- **Moderate**: Less than 75%
- **Strong**: At least 75%

To properly link an object to a user, our utility function must be able to take all 8 characteristics into consideration. Therefore, this function has many different networks, each one trained to identify a different characteristic of the FSLSM. With 8 characteristics and 4 different possible levels for each characteristic (not relevant, weak, moderate and strong), we would have a total of 32 networks, however, if a user is not a member of the first 3 levels, then he must be of the last one, what means that we do not need to check the last level, bringing the total to 24 networks.

It is important to note that there is no procedure (at least we were unable to find one) to determinate how to properly measure these aspects from the FSLSM. Therefore,

1 The complete implementation of the entire process can be found at: `https://github.com/MiguelAlvim/Master_Theseis_Material`

```
Given a MADALINE network with 1 hidden Layer, a small α learning rate and a
  stopping condition of 100% hit rate or 1000 iterations:
Consider N_i.bias as the neurode bias
Consider N_i.incoming_j as the weights from the neurode N incoming connection J
Consider N_i.value as the value obtained from all the received signals multiplied by
  all the weights plus the bias
Consider N_i.result the result from the neurode activation function, executed with
  the neurode N_i.value
while stopping condition is false do
    for each entry in the training set do
        Run the network and obtain a output. Output is valued into -1 or 1 (False
          or True);
        if Output equals to the expected class of the input then
            do nothing;
        else
            if Output equals to -1 then
                /* At least one of the neurodes should give a 1, change
                   the neurode N that is closes to obtain the correct
                   output.                                               */
                N = hidden layer neurode with the N.value closest to 1;
                N.bias = N.bias + α*(1 - N.result) ;
                for each incoming link j in the N neurode: do
                    N.incoming_j = N.incoming_j + α*(1 - N.result)*N.value ;
                end
            end
            if Output equals to 1 then
                /* All neurodes with should give -1, so we change all of
                   them that returned a 1                                */
                for each neurode N_i that has N_i.result = 1 do
                    N_i = hidden layer neurode with the N_i.value closest to 1;
                    N_i.bias = N_i.bias + α*(1 - N_i.result) ;
                    for each incoming link j in the N_i neurode: do
                        N_i.incoming_j = N_i.incoming_j + α*(1 - N_i.result)*N_i.value ;
                    end
                end
            end
        end
    end
end
```

Algorithm 1: MADALINE Rule 1 Training (MR1) for a OR gate

aside from the recommendations from the ILS, which we could not use in their totality due to the different scoring scenario, we had to create these divisions and networks based on our own opinions and observations.

We can summarize our ANN with:

- The MADALINE networks build by us have 1 hidden layer composed of 6 neurodes.
- They receive 8 inputs (all the characteristics of the FSLSM).
- They have 1 output neurode with a OR gate, capable of outputting 0 (false) or 1 (true).
- They are trained with our implementation of the MR1, shown on the algorithm 1, with the same stopping criteria.
  - Our learning rate α is *0.05*.
- As per the standard of the MR1, the weights of the links between the hidden layer and the output layer are fixed values. After testing we opted for using 0.2 on all of them.

4.2.3.2.1 Process of Recommendation

Our networks are trained with a synthetic database that has all possible types of objects based on the method created by Anitha and Deisy (2015).

For recommendation, a vector with all the 8 characteristics of the object is used as input for all the networks. Seen that a characteristic can only belong to one of the 4 possible levels, they are first checked for Strong affinity, if a negative result is returned, the network for the next level is used until we reach the last level (not suitable), where the level is automatically given to that characteristic.

Once all 8 characteristics are measured, in a similar process to the vector distance function, we score each result (8 results, with each one being a strong,moderate,weak or not relevant), with:

- Strong characteristics gain 2 points.
- Moderate characteristics gain 1 point.
- Weak characteristics gain 0.5 points.
- Not Relevant characteristics gain 0 points.

The final object score is calculated by multiplying all the scores by the user ILS results (transformed to percentage, as mentioned on this chapter) and adding the results together.

This process is repeated for all objects and once done, they are ordered in a vector in descending order, allowing the user to pick the N most recommended objects by getting the first N objects from the vector.

4.3 OBTAINING THE OBJECTS AND USERS

We made use of 4 different data sets for our experiments, 2 data sets with artificially created data and 2 data sets with real world data.

4.3.1 Artificial Data Sets

The artificial data sets were created with all the possible permutations from the ILS (for the artificial user data set) and all the possible permutations from the covered characteristics of the IEEE-LOM from Anitha and Deisy (2015).

With 5 possible results for each axis of the ILS (strong or moderate on the left, strong or moderate on the right or neutral) and with 4 axis, we have a total of $5^4 = 625$ combinations of possible students for our data set.

Using the method from Anitha and Deisy (2015) we have 6 relevant fields in the IEEE-LOM model. That give us: $2*8*3*9*5*2 = 4320$ possible learning objects.

Field	**Possible Values**
Field 1.7 - Structure	2
Field 4.1 - Technical Format	8
Field 5.1 - Interactivity Type	3
Field 5.2 - Learning Resource Type	9
Field 5.3 - Interactivity Level	5
Field 7.1 - Relationship Kind	2

Tab. 1 – Relevant fields of the IEEE-LOM from the Anitha & Deisy Method

This was done to guarantee that these data sets would cover all possible situations that can be found by these recommender systems. In the case of the neural network training this is particular useful and important towards avoiding conditioning the network for a particular subset of all possible cases[2].

4.3.1.1 Real data Sets

The real data sets were obtained from the works of Machado, Barrére, and Souza (2019)[3] and are composed of 101 students and 300 learning objects.

[2] These Synthetic data sets can be found at `https://github.com/MiguelAlvim/Master_Theseis_Material`

[3] These real data data sets can be found at `https://github.com/ufjf-dcc/LAPIC3-benchmark/tree/master/LaboratoryExperiments/Databases`

5 EXPERIMENT AND ANALYSIS

In this work we sought to test the following:

- Quality of our synthetic data sets in terms of its capabilities to represent a real world scenario.
- Capacity of the network to properly classify the learning objects according to Anitha and Deisy (2015).
- Is our utility function and therefore recommender system capable to differentiate our objects and recommend them?
- Is our metrics and overall modelling of our objects and users good/sufficient for a proper recommendation?

As a relevant factor, we would like to mention that all experiments were carried out in Python 3.7.4 using code written by us. All real numbers were represented by Python's '*float*' data type, wich has a 53b precision (16 digits - the usual '*double*' precision on other languages).

5.1 QUALITY OF SYNTHETIC DATA

To measure our synthetic data validity we compared the distribution of the entries presented in a real world data set with our created data set.

5.1.1 User data

As stated on the chapter 4, there are 5 possibilities when using the ILS classification (Strong Left [-11,-9], Moderate Left [-7,-5], Balanced [-3,3], Moderate Right [5,7] and Strong Right [9,11]). To easily visualize these we adopted a 0 to 4 classification system. Basically, Strong Left is 0 and as the classes go from left to right we increase the number by 1 until we reach Strong Right with a 4. A Object composed of 4 axis therefore has 4 numbers attached to it. For example, an object that is strong Left on all 4 axis would be seen as [0,0,0,0]. This characteristic allows us to show the density of user types on a data set in a single 2 dimension graphic, as seen on figures 5 and 6, with the first entry being [0,0,0,0] and the 625th entry being [4,4,4,4].

Our synthetic data has a single example of all possible types predicted into the ILS. Therefore a homogeneous distribution is expected.

	STRONG	MODERATE	BALANCED
ACTIVE	125/625	125/625	375/625
REFLECTIVE	125/625	125/625	375/625
SENSING	125/625	125/625	375/625
INTUITIVE	125/625	125/625	375/625
VISUAL	125/625	125/625	375/625
VERBAL	125/625	125/625	375/625
SEQUENTIAL	125/625	125/625	375/625
GLOBAL	125/625	125/625	375/625

Tab. 2 – Concentration of Characteristics of singular ILS classes on the Syntetic data set

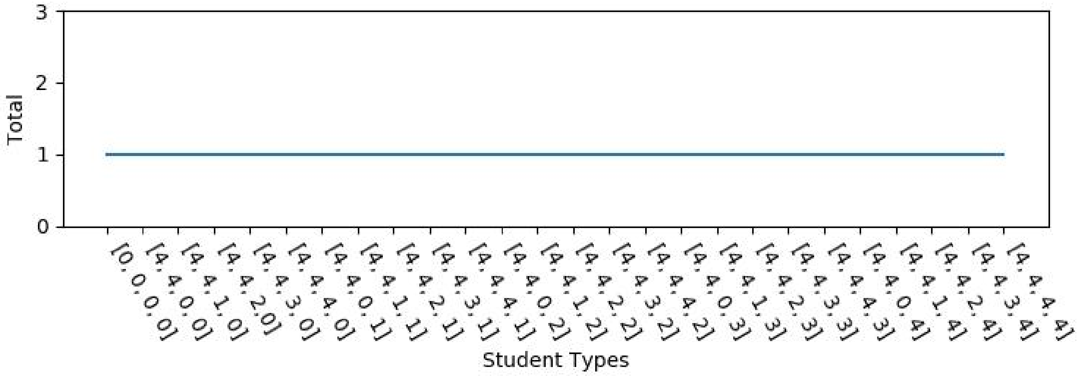

Fig. 5 – Distribution of User types on the Synthetic data set

Our real world data set however was obtained from students of a computer science bachelor degree class and has less than 1/6th of the entries of our synthetic base. It is expected that it will reflect the tendencies of that subgroup of students with a possible bias towards certain characteristics of the FSLSM, as is known that different bachelor degrees and academic fields tend to attract different types of people with different preferences.

ACTIVE	2/101	20/101	79/101
REFLECTIVE	4/101	10/101	87/101
SENSING	12/101	34/101	55/101
INTUITIVE	1/101	3/101	97/101
VISUAL	10/101	29/101	62/101
VERBAL	3/101	9/101	89/101
SEQUENTIAL	1/101	26/101	74/101
GLOBAL	2/101	7/101	92/101

Tab. 3 – Concentration of Characteristics of singular ILS classes on the Real data set

By comparing both data sets we can see that the synthetic data set has entries capable of representing all members of our real world data. The concentration of entries however differs, with the real data set demonstrating a higher tendency towards a Balanced and Moderate style for most students than in our balanced synthetic scenario. This difference will not interfere with any results for our utility functions, as they focus on the characteristics of our learning objects for the recommendation (by matching then with the users).

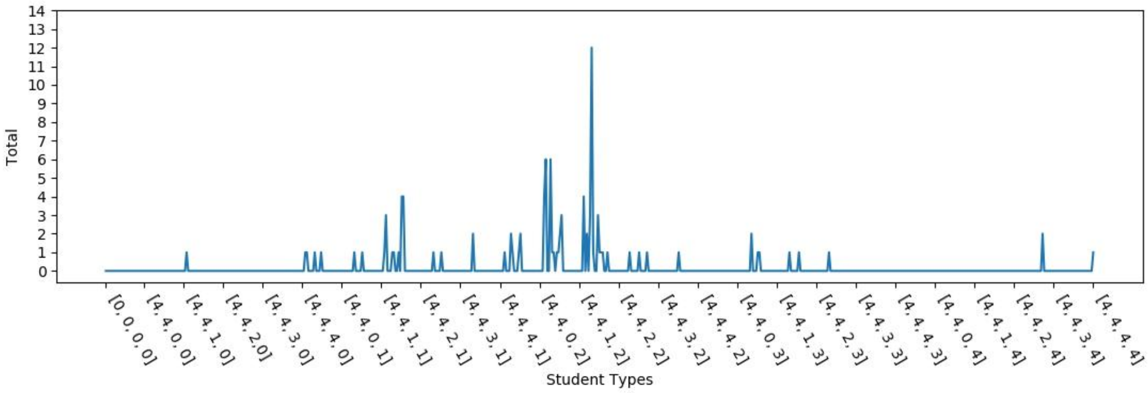

Fig. 6 – Distribution of User types on the Real data set

5.1.2 Object Data

Our real world data set is composed of a collection of 300 digital multimedia objects and tests. This particular case is actually rather common on the majority of digital learning object repositories.

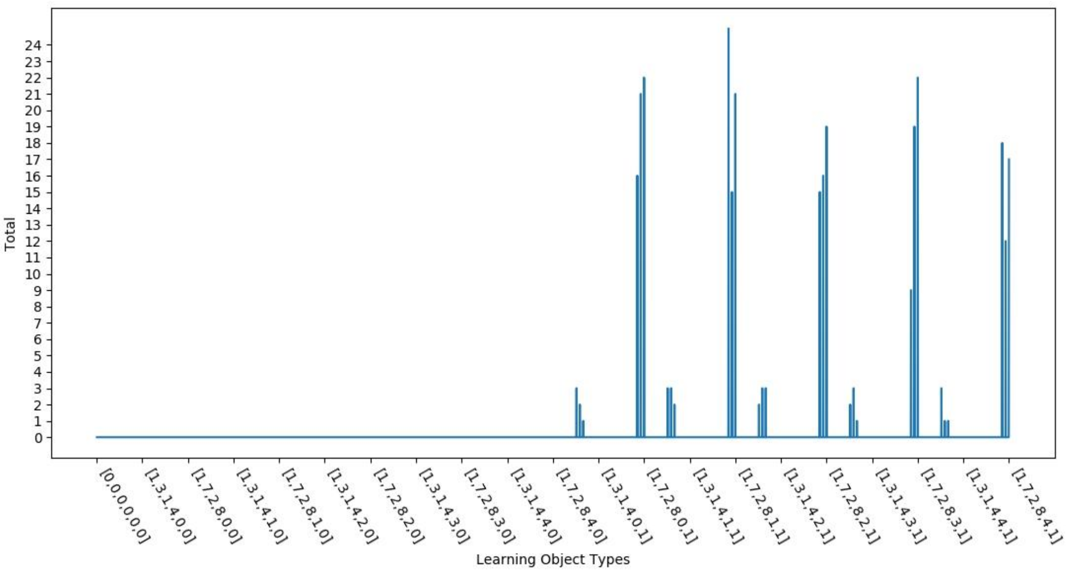

Fig. 7 – Distribution of Learning Object types on the Real data set

Our synthetic learning object data follows the same principle that our synthetic student data has: One member for each possible result from our classification method with the learning objects that would be Anitha and Deisy (2015) automatic method.

As we did with our users data sets, we can use a single vector to represent a object, but instead of 4 values ranging from 0 to 4, we'll have the 6 explained in table 1 written on chapter 4.

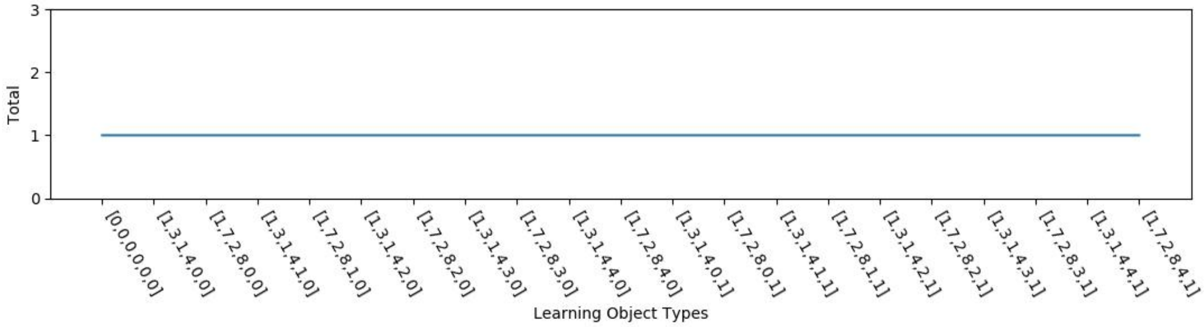

Fig. 8 – Distribution of Learning Object types on the Synthetic data set

We can observe a very big difference in the concentrations of learning object types from our synthetic data set and our real world example. As previously discussed, it is fair to consider this real world data set to be a decent representation of the types of learning objects found on a digital learning object base, meaning that the use of our synthetic base can actually generate different results than the use of our real world data set on the process of training our ANN for one of our utility functions. That possibility is investigated and analysed more on the next part of our experiment.

That said, our synthetic data was shown to have a element capable of represent any element in our real world example, what means that it is capable to represent a real scenario when needed.

5.2 NETWORK CLASSIFICATION AND THE IEEE-LOM TO FSLSM METHOD

As stated in chapter 4 we created 24 ANN to be able to classify all 4 different types for each axis on our learning objects. Following our conclusions, we decided to use our synthetic data set of learning objects, to guarantee that, in theory, our ANN would be able to work with any possible learning object that exists.

Each Network was trained until it was able to obtain 100% hit rate or reached 1000 iterations, using the algorithm 1. In the cases of non 100% hit rates, the best result from the training was used (the one that obtained the highest success/hit rate).

Although in principle unnecessary, networks that classify the lack of relevancy were created for us to obtain a complete picture of the capacity of our network and classification method.

In overall, the networks using our implementation of the MR1 were able to completely classify our learning objects on most of the existing characteristics of the FSLSM. However some cases obtained a sub-par classification rate, with some characteristics having 2 of their ANN not reaching 100% hit rate, what makes a 100% accurate identification by the ANN system impossible in those cases.

ANN	HIT (%)	ANN	HIT (%)
ACTIVE STRONG	100	VISUAL STRONG	100
ACTIVE MODERATE	57.92	VISUAL MODERATE	99
ACTIVE WEAK	67.71	VISUAL WEAK	100
ACTIVE NOT_RELEVANT	100	VISUAL NOT_RELEVANT	100
REFLECTIVE STRONG	100	VERBAL STRONG	100
REFLECTIVE MODERATE	57.82	VERBAL MODERATE	97
REFLECTIVE WEAK	72	VERBAL WEAK	100
REFLECTIVE NOT_RELEVANT	100	VERBAL NOT_RELEVANT	100
SENSING STRONG	100	SEQUENTIAL STRONG	100
SENSING MODERATE	100	SEQUENTIAL MODERATE	82.06
SENSING WEAK	100	SEQUENTIAL WEAK	100
SENSING NOT_RELEVANT	100	SEQUENTIAL NOT_RELEVANT	100
INTUITIVE STRONG	100	GLOBAL STRONG	100
INTUITIVE MODERATE	100	GLOBAL MODERATE	100
INTUITIVE WEAK	100	GLOBAL WEAK	100
INTUITIVE NOT_RELEVANT	100	GLOBAL NOT_RELEVANT	100

Tab. 4 – MADALINE ANN Synthetic Training Results

The process of training a ANN can be rather laborious, with cases where many tweaks become necessary to achieve a good hit rate. In our case, we attempt to temper with values such as the learning rate α, the number of hidden neurodes, and even the weights of the connections between the hidden layer and the output layer that, for the MR1 rule, are fixed. None of those alterations brought better results.

After that, instead of further modify our ANN, we focused on the data sets. Our earlier experiments demonstrated that they indeed have a representation of all the objects that exists in a real data set, but that is considering the IEEE-LOM model and we are using the objects that were transformed to the FSLSM.

By analysing the method used by Anitha and Deisy (2015) and checking the weights of each of the links from each data set, we can clearly see that some of the characteristic of the IEEE-LOM used on the method have effect on more than 1 characteristic of the FSLSM. This means that some elements of our learning objects, although not connected by the same axis, are connected by the same characteristics, what directly interferes with our capability of learning and classifying our learning objects for our recommendation.

We can synthesise these findings in the following way:

- Using the method of adaptation for the IEEE-LOM to the FSLSM we end up with characteristics from different axis having some degree of correlation between each other.
  - This is a good thing in principle, as it helps with the ANN learning process.

- However not all 8 characteristics have sufficient correlation with the rest of the model, what means that to properly classify all 4 classes created by us, or even the original 3 from the ILS, the ANN can only really use the individual value from that characteristic alone.
  - That becomes a problem giving our classification method of a OR gate. For the Strong and Not Relevant extremes is actually a very simple process, just give a lot of weight on the input of that characteristic in one neurode and the value from that neurode will guarantee a correct output. Make the inverse on the case of the non relevancy with all the inputs and you'll also be able to get a 100% hit rate. In fact, that is what we observed on the results from the Active Strong and Not Relevant networks.
  - For the middle classifications, such as Weak or Moderate, we must be able to detect when the input is high enough to be relevant but not too high to be too strong. Then, we get into the problem that if we only use one characteristic, we cannot not detect if an input is in the Weak or Moderate state without also getting a positive answer for the class that is higher, because if our weight were able to elevate the input signal to the point where it became strong enough to be detect as a "Weak" level signal, then stronger signals would become even stronger and also trigger the positive output. In straighter terms:

    A "True" output from our ANN comes from the fact that at least one neurode from the hidden layer answered 1 ("True"). Seen that most of the inputs are irrelevant, that means that one neurode must increase the relevant signal to the point were it becomes at least ≥ 1.

    In a simple multiplication operation, if you are able to increase a positive value to be at least 1, then any value that is bigger than that value will also be increased to be bigger than 1.

    That means that is impossible for the neurodes to use a single input in this scenario to identify values that are stronger than a threshold but also weaker than another threshold. There is just not enough information to work with given the method that they use to classify.
- We must conclude that it is not possible to use a OR gate MADALINE with the results from the method proposed by Anitha and Deisy (2015) to obtain 100% accuracy on all possible learning objects that are sorely defined by the results from the Method. There is not enough correlation between the characteristics to work with.

As a last action to verify if this conclusion would hold true on real world data, we also performed the same training process using the 300 real world data from Machado, Barrére, and Souza (2019). The results were better, with more characteristics of the FSLSM achieving 100% success. The ones that had lower rates with our synthetic data however, kept their overall rates even with this data set. The results of the training can be found on table 5.

That shows us that even with a much smaller and real set of learning objects, the observed problems persisted. That is indeed an inherent characteristic of the method used.

ANN	**HIT (%)**	**ANN**	**HIT (%)**
ACTIVE STRONG	100	VISUAL STRONG	100
ACTIVE MODERATE	60.67	VISUAL MODERATE	100
ACTIVE WEAK	69	VISUAL WEAK	100
ACTIVE NOT_RELEVANT	100	VISUAL NOT_RELEVANT	100
REFLECTIVE STRONG	100	VERBAL STRONG	100
REFLECTIVE MODERATE	55.67	VERBAL MODERATE	100
REFLECTIVE WEAK	67	VERBAL WEAK	100
REFLECTIVE NOT_RELEVANT	100	VERBAL NOT_RELEVANT	100
SENSING STRONG	100	SEQUENTIAL STRONG	100
SENSING MODERATE	100	SEQUENTIAL MODERATE	100
SENSING WEAK	100	SEQUENTIAL WEAK	100
SENSING NOT_RELEVANT	100	SEQUENTIAL NOT_RELEVANT	100
INTUITIVE STRONG	100	GLOBAL STRONG	100
INTUITIVE MODERATE	100	GLOBAL MODERATE	100
INTUITIVE WEAK	100	GLOBAL WEAK	100
INTUITIVE NOT_RELEVANT	100	GLOBAL NOT_RELEVANT	100

Tab. 5 – MADALINE ANN Real Data Training Results

It is fair to note that the method proposed by Anitha and Deisy (2015) also used the classification of called human “experts” on the characteristics of the FSLSM to classify the learning objects and that could skewer the LO towards classifications that are better suited for our ANN. That said, the method does allow for the complete use of the automatic IEEE-LOM based procedure and we know that for the real life scenario, it is unreasonable if not logistically impossible to manually and properly classify all learning objects from every, digital or not, data set.

5.3 UTILITY FUNCTIONS RESULTS

This last experiment focused on checking if we could get enough differentiation from every learning object using our utility functions.

5.3.1 Vector Similarity

We ran our vector similarity recommender function using our entire synthetic and real data sets of learning objects to recommend to all the 101 students from our real student data set.

Since we could not run these experiments with the real world students, only with their profiles, we unfortunately could not use the metric of the users' opinion on the results from our system. Therefore, to measure any sense of success we instead analysed the capacity of differentiation from all the objects and users among each other. In other terms, we are checking if, among all the learning objects and our students, substantially different results were given to substantially different students and vice-versa or, if not, if the results both obtained have characteristics that appeal to all cases.

To differentiate our students we used the same vector distance metric (without the α modifier) and compared each student with all the other ones. The highest difference was found between the $27th$ ([4,4,4,4] type with a [11,11,11,11] ILS score) and the $49th$ ([1,0,0,1] type with a [-7,-9,-9,-5] ILS score) student, with a similarity factor of **2.388029375866218** (as said on chapter 4, the higher the less similar) out of possible maximum of $\sqrt{8}$ or approximately **2.8284271247461903**.

Before starting to demonstrate our experiments and findings, it is important to note that ideally, we should run this experiment not only with the extremes of each part of the learning styles (like we did) but also with other pairs (preferably randomly selected) from our 101 students data set. However, limitations made us to work with only the pair of extreme cases.

Using the Real data set of 300 objects, we compared the most well recommended objects (similarity < 0.4) for both users ; The **$27th$ user had 70** objects and the **$49th$ had 87**. In this subgroup, **zero objects were recommended for both users at the same time (no intersection exists in the results)**, what give us the indication of a good differentiation from our utility function. That said, once we verify the individual score for the objects, we can see that several ones (more than 10) obtained the exact same scores, and this case is true in both users. On the $27th$ user, the most well recommended objects, 17 in total, all had the score of **0.10539871915730283**.

After verifying the type of the objects, we realized that all of them were videos with the same types of the relevant IEEE-LOM tags, what helps to indicate the possible low ability of the method created by Anitha and Deisy (2015) or of the FSLSM to differentiate the learning objects enough for our recommender needs, as both approaches do not use the content of the video itself for further classification (what cannot be completely blamed on them, ans the IEEE-LOM itself also do no go too deep into those individual qualitative merits of the media it classifies).

The same process was done for our synthetic data set and similar results were obtained, with **526 objects recommended for the $27th$** and **455 for the $49th$ user** and **zero intersection from both groups**, what reinforces the results and conclusions obtained from our real data set experiment.

5.3.2 Artificial Neural Network

Following the same principles from our test with the Vector Similarity function, we ran our ANN recommender function with all the 101 students from our real student data set and checked the results with both the synthetic and real data sets.

On the same merits from the vector experiments, we opted to use the $27th$ and the $49th$ students for our comparisons.

Our ANN recommender process generates smaller variations on the score of each possible object, with the maximum possible being 8 and the minimum 0. Due to this smaller granularity given by the ANN, we obtained much larger amounts of objects receiving the same score .

First we compared the results from our real world data set of learning objects and decided to get the objects that received the highest score for each user. We opted for this approach instead of the threshold used on the vector function due to the large amount of objects receiving the same score; on the case of the **$27th$ student we had 70** objects with a score of 8 and for the **$49th$ student we had 77** wth a score of approximately 7.7273 **with zero intersections**.

With the synthetic data set we had a situation where our $27th$ student receive 656 recommendations with the highest score of 7.7273 and our $49th$ student 'only' 182 with the highest score of 8. In these groups there were no intersections as well.

We boosted the amount of the $49th$ by adding the next most well recommended objects (all objects above the score of 6.9091 were added). That gave us 612 objects and still, no intersection with any objects from our $27th$ student.

Although the objects receive less differentiation than with our vector utility function approach we still were able to differentiate our objects qualities enough when faced with diverse preferences from our students. This gives us leeway to consider that the ANN approach can definitely achieve good results, specially considering that our ANN receive an overall simple and naive implementation.

6 CONCLUSIONS

In our research, we sought to explore the use of Learning Styles as the main characteristics of a multiple-criteria Recommender System on virtual learning objects databases. Our main motives were to explore the said capabilities of LS as indications of relevant characteristics of student learning preferences and habits.

Learning styles are usually treated as an easily obtained information, mostly through a simple questionnaire (ALVIM DE ALMEIDA; BARRÉRE; SOUZA, 2019), which could definitely help any recommender system to avoid problems such as “cold starts”.

During our work, we manage to create a complete data set of synthetic data that was shown to have objects capable of representing all possible types of students, in accordance with the FSLSM. The same is also true for the creation of a synthetic data set of learning objects based on the IEEE-LOM.

Another set of positive results and contributions obtained was the completely transparent implementation of a ANN based recommender system (with all the details and algorithms for its creation being discussed and explained). Albeit simple in design, our perceived lack of details about the workings of other ANN based systems (HAMADA; HASSAN, 2018, and others not directly cited here), led us to conclude that, by the lack of other examples, this aspect of our production can be better used, scrutinized and tested than the other implementations that we encountered, and that itself can better lead towards more robust works by us or others.

In our experiments we were able to detect several limitations on the proposed method by Anitha and Deisy (2015) in terms of its the capability of a granular and precise learning object identification and classification withing the FSLSM. If that is due to the limitations of the method or of the FSLSM itself we cannot say with an absolute amount of precision, but considering that the method was utilizing a small subset of information available on the IEEE-LOM and proposed to mix its automatic results with human-made classifications (not a viable option for many cases), we believe that is fair to conclude that, perhaps, there is no completely precise method currently available that maps learning objects towards the FSLSM in a totally satisfactory manner for a recommender system. The adoption of another method to classify learning objects on the FSLSM or even methods to map students and learning objects into another type of classification system that properly suits the process of recommendation is definitely a possibility that needs to be studied further.

In fact, we can only conclude that learning styles themselves are not enough to currently create a substantial level of differentiation of learning objects. Granted that perhaps the method chosen to bridge the gap between LO and LS was not the most effective, as discussed on the paragraph above, we must also remember that a LO is much more than just the way that it interacts with the student. Learning styles alone do not approach other very important characteristics of LO, such as their specific contents, the level of difficulty, the length of time necessary to properly absorb it, the fact that it requires one or maybe several students to be properly used and so on. Also, in the case of digital LO we must be aware of user contexts such as location, internet access, download/upload speeds, data caps, device characteristics among others.

A real and proper LO recommender system should tackle all those factors to be able to attend to all possible students needs. In fact, as is shown in the literature, the more is known about a object, the better one can accurately recommend it. Perhaps working with these information and with the LS we could create a much more precise recommender system for learning objects. That is definitely a possible avenue for future works.

Another avenue for more works would be to finish the testing procedures of our recommender system, by making more comparisons with other pairs of students, as commented on chapter 5. Also analysing the distribution of styles on other different groups of students, as our 101 were all computer science students, might too give more insights on how to tailor these systems.

That all said, we do believe that the use of learning styles as complementary information or as a starting point have it's merits, as we were able to detect very different users and still recommend them objects that in theory would suit them. Still, perhaps our approach wasn't the best possible to demonstrate it. ANN on recommender systems require quite a bit of preparation and planing to be properly used, as we hopefully demonstrated with our subpar results. As we finish this work, we cannot dismiss that Gediminas and YougOK were right when they did their call for exploring new possibilities in recommendation (ADOMAVICIUS; KWON, 2015), as ANN have many aspects that need to be properly studied in many different scenarios, what requires many different and individual studies to be made, before we can take any definitive conclusion on their capabilities for recommendation. We hope that this research was able to add a new perspective on what needs to be done in this aspect.

References

ADOMAVICIUS, Gediminas; KWON, YoungOk. Multi-Criteria Recommender Systems. In: RECOMMENDER Systems Handbook. New York: Springer Science + Business Media, 2015. DOI 10.1007/978-1-4899-7637-6_25. chap. 25.

ADOMAVICIUS, Gediminas; TUZHILIN, Alexander. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. **IEEE Transactions on Knowledge & Data Engineering**, IEEE, n. 6, p. 734–749, 2005.

ALVIM DE ALMEIDA, Miguel; BARRÉRE, Eduardo; SOUZA, Jairo. Learning Style identification and usage in academia: A systematic mapping. In: 1. BRAZILIAN Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). [S.l.: s.n.], 2019. v. 30, p. 1341.

ANITHA, D; DEISY, C. Proposing a novel approach for classification and sequencing of Learning Objects in E-learning systems based on learning style. **Journal of Intelligent & Fuzzy Systems**, IOS press, v. 29, n. 2, p. 539–552, 2015.

BAEZA-YATES, Ricardo; RIBEIRO-NETO, Berthier, et al. **Modern information retrieval**. [S.l.]: ACM press New York, 1999. v. 463.

COFFIELD, Frank et al. **Learning styles and pedagogy in post-16 learning: A systematic and critical review**. [S.l.]: Learning and Skills Research Centre London, 2004.

FELDER, Richard M; SILVERMAN, Linda K. Learning and teaching styles in engineering education. **Engineering education**, v. 78, n. 7, p. 674–681, 1988.

FELDER, Richard M; SOLOMAN, Barbara A. Index of learning styles questionnaire. **Retrieved July**, v. 20, p. 2009, 1997.

HAMADA, Mohamed; HASSAN, Mohammed. Artificial neural networks and particle swarm optimization algorithms for preference prediction in multi-criteria recommender systems. In: MULTIDISCIPLINARY DIGITAL PUBLISHING INSTITUTE, 2. INFORMATICS. [S.l.: s.n.], 2018. v. 5, p. 25.

HASSAN, Mohammed; HAMADA, Mohamed. Enhancing learning objects recommendation using multi-criteria recommender systems. In: IEEE. 2016 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE). [S.l.: s.n.], 2016. p. 62–64.

________. Performance analysis of neural networks-based multi-criteria recommender systems. In: IEEE. 2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE). [S.l.: s.n.], 2017. p. 490–494.

HASSAN, Mohammed; HAMADA, Mohamed. Performance comparison of feed-forward neural networks trained with different learning algorithms for recommender systems. **Computation**, Multidisciplinary Digital Publishing Institute, v. 5, n. 3, p. 40, 2017.

IEEE. **1484.12.1-2002 - IEEE Standard for Learning Object Metadata**. [S.l.: s.n.], Sept. 2002. Permanent link: https://ieeexplore.ieee.org/servlet/opac?punumber=8032.

JUNG, C.G. **Psychological Types**. Princeton, N.J: Princeton University Press, 1971. (Originally published in 1921.)

KIRSCHNER, Paul A. Stop propagating the learning styles myth. **Computers & Education**, Elsevier, v. 106, p. 166–171, 2017.

KOLB, D.A. **Experiential Learn-ing: Experience as the Source of Learning and Development**. Englewood Cliffs, N.J: Prentice-Hall, 1984.

MACHADO, Marcelo de Oliveira Costa; BARRÉRE, Eduardo; SOUZA, Jairo. Solving the Adaptive Curriculum Sequencing Problem with Prey-Predator Algorithm. **International Journal of Distance Education Technologies (IJDET)**, IGI Global, v. 17, n. 4, p. 71–93, 2019.

PHILLIPS, Jon. Are “Learning Styles" Real? **Dialog on Language Instruction**, v. 29, n. 1, p. 32–35, 2019. From the Defense Language Institute: Foreign Language Center, Presidio Monterey.

PIERRAKEAS, Christos et al. The Variability of the Reasons for Student Dropout in Distance Learning and the Prediction of Dropout-Prone Students. In: MACHINE Learning Paradigms. [S.l.]: Springer, 2020. p. 91–111.

WIDROW, Bernard. Generalization and information storage in network of adaline 'neurons'. **Self-organizing systems-1962**, Spartan Books, p. 435–462, 1962.

WIDROW, Bernard; LEHR, Michael A. 30 years of Adaptive Neural networks: Perceptron, Madaline, and Backpropagation. **Madaline and Backpropagation Proceedings of the IEEE**, v. 78, n. 9, p. 1415–1441, Sept. 1990.