

RS-REJA (Recommender System of REstaurants in JAén) is the result of the research project:

Recommender System of Restaurants based on Fuzzy Logic (JA031/06)

granted by the Office of Turism, Commerce, and Athletics of the Andalusian Government (Consejería de Turismo, Comercio y Deporte de la Junta de Andalucía) as part of the 2006 official programme *Subsidies for the development of campaigns to raise public awareness of the touristic quality of Andalusia*

The purpose of this research project is the development of a recommender system of **Restaurants in the province of Jaén** based on soft computing techniques that assist the calculations of the recommendations.

In order to learn how **RS-REJA (REJA** from now on) works, please, continue reading.

The system displayed here implements two recommender types:

1.

Collaborative Recommender System

2.

Knowledge-based Recommender System

Both types are based on the techniques examined in the articles on **Recommender Systems and Soft Computing Techniques**

published by the members of the

Intelligent Systems

Research Group of the University of Jaén.

REJA is a highly functional recommender system implemented upon a database of restaurants in the province of Jaén. This database, however, has no true but feigned data of the restaurants, as the span of time for the implementation of the project and the subsidy granted by the Andalusian Government make difficult the gathering of such data. Nevertheless, we are

currently leading our efforts to get the chance to work with true and reliable data in the near future.

Functionality

As we have mentioned above, the recommender functions of **REJA** have been completely developed and implemented, even if the data used are not real-valued components.

Most recommender systems usually employ databases where the users rate the items on a numeric scale. However, on many occasions the users may feel this method lack insight. As **REJA** works upon a database of restaurants, to score this type of objects numerically may seem inadequate. This has led us to work on **linguistic variables** through a computational model of [2-Tuples](#) in order not to lose information in relation to the use of numerical variables.

REJA includes the two recommender systems mentioned before:

Collaborative System

In a **Collaborative Recommender System** the recommendations are exclusively based on the resemblance terms among the users. The operation of this type of recommender system runs as follows:

1.

The system saves a **profile for each user** which includes the database items that the user knows and evaluates.

2.

The **degree of resemblance among the different users** is measured on the basis of their profiles. Hence, groups of users with similar characteristics are created.

3.

The system will use all the information obtained in the previous actions in order to **make the recommendations**

. It will recommend to each user items they have not evaluated and that have been positively evaluated by the rest of the members of the group.

Consequently, these recommender systems do not take into account the contents and characteristics of the products they recommend, but the taste of those users who share a similar profile with the user who requires the service.

To obtain a top-quality Collaborative Recommender System it is essential to use good Collaborative Filtering Algorithms. These algorithms can be classified into two categories:

-

Memory-based or User-based Algorithms -> These algorithms employ the whole database for the generation of predictions and recommendations. They are very popular, but they are likely to suffer from scarcity in scalability. (o “suffer from problems of scarcity and scalability”)

-

Model-based or Item-based Algorithms -> In order to make recommendations, these algorithms first develop a model of the database by using rules, Bayes, clusterings...

Collaborative Filtering Algorithms keep to the following steps:

1.

The database is divided into two disjunctive conjuncts: **training** and **test**.

2.

k-Nearest-Neighbour algorithms, or those similar to every single item, are calculated upon the training conjunct by using any resemblance metrics.

3.

Predictions for the users of the test conjunct are estimated by using any prediction algorithm.

4.

The **precision** of the obtained predictions is measured.

In **REJA** we have not implemented any collaborative filtering algorithms, but we have used a **collaborative filtering engine**

called

[COFE](#)

, and we have worked on the obtained results.

Finally, here is a brief visual example of how the Collaborative Recommender System implemented by REJA works:

{youtube}ZyiuWp6HBNM{/youtube}

Knowledge-based Recommender System

The most commonly used recommender systems (collaborative and content-based) have a series of constraints that can have an influence on the quality of the generated recommendations. These recommender systems need historical ratings (or “information”) of the past actions of the users in order to generate recommendations. However, this information is seldom useful, adequate, or available, as the circumstances of the users may vary: we may find people who visit the city for the first time –and, therefore, it is the first time they use our system–; users who, in spite of having used this system before, wish to have lunch in a specific restaurant to mark a special occasion (birthday party, retirement celebration...) unrelated to any of their past actions; or users who, even though they have interacted with the system, have not provided enough information to allow the collaborative system to make accurate recommendations.

Knowledge-based Recommender Systems arise with the aim of making recommendations in this type of situations. To do so, these systems employ, by means of a Case-based reasoning, the knowledge of the preferences provided by the user (user’s knowledge), the knowledge of the database items that the system has (catalogue knowledge), and the knowledge of how these items meet the requirements of the users (functional knowledge).

In **REJA** we have implemented a Knowledge-based Recommender System which consists of three different stages:

1.

Capturing knowledge of user preferences: the system must gather information about the preferences of the user in the most precise and simple way. In these systems, this information is usually provided by means of an example, although the user may be allowed to provide information manually. In **REJA** this information may be provided in three different ways:

1.

By means of an example: the user presents a restaurant which should serve as an example of what they need. This option will be particularly useful to those users who know some restaurants in the area.

2.

By establishing the search options (or “search parameters”): in this case the user expresses their preferences considering *menu price, type of cuisine*, and *type of restaurant*. This is the best option in case the user does not know the area.

3.

By providing a set of preferences for a group of given restaurants: This option will only be available to registered users. The user will have to choose a restaurant to their liking and the system will show three more restaurants (two restaurants previously evaluated by the user and another restaurant diametrically opposed to the selected one). Next, the user will

have to describe the preferences for the restaurant they have selected and for the restaurants chosen by the system. On the basis of these preferences, the system will assemble a [preference matrix](#) which will be essential to compute the user's **partial profiles** (by using the [IOWA](#) aggregation operators) and their final profile.

1.

Filtering database items: the aim of this phase is the filtering of restaurants so that we only deal with those restaurants which may best meet the user's preferences. The filtering is made by using the *price and type of cuisine* attributes.

1.

Recommendation: Among all the restaurants that to a large extent may suit the user's preferences, those that best meet their demands must be chosen. In order to do so, the group of restaurants obtained in the previous phase will be arranged according to their *quality* and *services* through a 2-Tuple weighted average, and the 10 best valued restaurants will be displayed.

To conclude, just a brief example to show how the Knowledge-based Recommender System acts with **non-registered users**:

{youtube}HzdR4EKEXs4{/youtube}

On the other hand, it runs as follows with **registered users**:

{youtube}UiXTbPDusUk{/youtube}

REJA staff would like to encourage you to register with us and benefit from the application. **WE LCOME AND ENJOY OUR SERVICES!**

Kindest regards.