Fuzzy Logic based Intelligent Farming Multi Agent System

Mrs. Anita Gutta, Dr. Priti Srinivas Sajja
Department of Computer Science & Technology
Sardar Patel University
Vallabh Vidyanagar, India

Abstract

A lot of tools have been developed to personalize content on the web. And filtering techniques like content based and context filters get the proper information to users. Information captured in user profiles have certain degree of imprecision and uncertainty. Fuzzy logic provides a means to overcome this short coming. This paper proposes a means to use Fuzzy Linguistic variables in user profiles and how to handle the vague and imprecise information.

Keywords - Intelligent Multi Agent systems, Fuzzy Linguistics, Information Retrieval system, Jade systems.

1. Introduction

The growing ball of information on the web needs more than a browser and good search engine to extract quality information from it. A lot of information on the web is designed for human consumption but computers are the means through which we get to the information. Computers are better at handling carefully structured and well-designed data, yet even where information is derived from a database with well-defined meanings, the implications of those data are not evident to a robot browsing the web [1]. Semantic web (SW) is a vision of next generation of World Wide Web (WWW) where information on web will be semantically annotated to be understandable by machines and crawlers. SW promises opportunities for exploitation of rich knowledge resources spread across WWW for processing, filtering and mining of knowledge by intelligent agents [2].

The Semantic web relies heavily on Ontologies and formal structure of data. Real world inputs are not always crisp and have a lot of vagueness in them. In order to handle this vagueness fuzzy logic is presented as a possibility. Specifically Fuzzy Linguistic variables will be used in the application proposed in this paper.

This paper is organized as follows: Section 2 goes over the technologies used in the paper, Section 3 goes over the proposed model, Section 4 describes the implementation details and Section 5 concludes the paper.

2. Technologies

Agents: An agent is an autonomous software unit that can exist independently of other similar units in the software system. An agent performs some functions for other agents or external actors. Agents communicate with each other via messages via an agent communication language. Agents react proactively to the changes in the environment. They are autonomous and very adaptive. Agents can be simple or complex based on the problem they are trying to solve. Different types of agents can be mixed up in the same application depending on the need. As per the Foundation for Intelligent Physical Agents (FIPA), the agent definition is: "An agent is the fundamental actor in a domain. It combines one or more service capabilities into a unified and integrated execution model which can include access to external software, human users and communication facilities [8]. It is just one way of defining agents. Universally accepted definition of agents is not possible because the attributes and characteristics of agents differ greatly from various domains.

Multi-agent system (MAS): It is a system composed of multiple interacting intelligent agents. Multi-agent systems are generally used to solve problems that are hard to solve with individual programs or agents. They offer a notably more powerful and natural abstraction for modelling and developing systems than conventional methods such as procedural abstraction, abstract data types and object oriented abstraction [10]. Agents also provide benefits of the conventional OO paradigm such as modularity and reusability. When a problem is too complex or unpredictable, the most effective way to address it is to develop a number of
modular agents, each of which specializes at solving a particular aspect of the problem [11].

**Fuzzy logic:** Fuzzy logic has already been implemented in a variety of systems across various domains and this technique has produced impressive results. It is very helpful where human like reasoning is required. Human intelligence and machine processing capabilities are possible with fuzzy logic. The term “fuzzy logic” emerged in the development of the theory of fuzzy sets by Lotfi Zadeh [9] in 1965. He modified conventional set theory in which an individual could have a degree of membership which ranged over a continuum of values, rather than being either 0 or 1. Fuzzy logic mainly focuses on quantifying vague or uncertain terms that appear in our natural language conversations. These terms are known as linguistic variables [9], or fuzzy variables. In addition to fuzzy linguistic variables there are also IF-THEN rules defined in the form of fuzzy rules. These will be used to transform the vague inputs into crisp outputs.

### 3. Proposal

The above discussed technologies intelligent multi agents and fuzzy logic are used in a web application and this combination greatly enhances user data retrieval process. It aptly captures user’s requirements, with the human like reasoning and usage of fuzzy linguistics produces and optimal system where even a user with minimum computer experience can benefit the most.

---

**4. Implementation**

The system is implemented using Jade [3]. The basic application is developed using intelligent multi agent systems. Specific Personalization techniques mentioned in [12] are applied using different agents developed with Jade. For the Fuzzy piece an open source library called jFuzzyLogic [4] is used. It is written entirely in java and is following the standard for Fuzzy Control Language (FCL) published by the International Electro technical Commission (IEC 61131-7) [5]. The web application that is proposed in this paper is specifically developed for farming system.

The customization of the application is done through user input captured in user profiles. The data in user profiles is captured when a new user is created. This user information can also be updated later when the user’s working environment changes. Also the feedback given by the user through various ratings and comments will be analyzed by the system and suggestions will be made to the user to better their search criteria and search results.

There is always a degree of vagueness and uncertainty when getting information from users using natural language conversations. The same words mean different things to different people and most of the times and it is difficult to translate the linguistics into crisp values. This is where fuzzy linguistics comes into picture.

---

![Diagram](image.png)
The two main pieces of data considered in this application as inputs to fuzzy system are the “size of the farm” of user and their “internet connection speeds”. There are lot of other input variables also which are not well defined but in the farming application based on these two input variables the output can be significantly different so these two are taken to show how the proposed system can work effectively. The classification values for farm scale as shown below are taken from the Ministry of Agriculture, Government of India [7].

Similarly different connection types are shown below.
A configuration file is maintained with all the different input values and rules. This fcl file is key to the Fuzzy system as it looks at the inputs passed on from user and applies the appropriate rules.

FUNCTION_BLOCK customfarming
// Define input variables
VAR_INPUT
    connectiontype: REAL; farmscale: REAL;
END_VAR

// Define output variable
VAR_OUTPUT
    portaltype: REAL;
END_VAR

// Fuzzify input variable 'connectiontype': {'dialup', 'isdn', 'dsl', 'cable', 'satellite'}
FUZZIFY connectiontype
    TERM dialup: = (0, 1) (20, 1) (56, 0);
    TERM isdn: = (50, 0) (100, 1) (128, 0);
    TERM dsl: = (128, 0) (4000, 1) (8000, 0);
    TERM cable: = (4000, 0) (10000, 1) (20000, 0);
    TERM satellite: = (1000, 0) (8000, 1) (16000, 0);
END_FUZZIFY

// Fuzzify input variable 'farmscale': {'marginal', 'small', 'semimedium', 'medium', 'large'}
FUZZIFY farmscale
    TERM marginal: = (0, 0) (0.5, 1) (1, 0);
    TERM small: = (0, 0) (1, 1) (2, 0);
    TERM semimedium: = (2, 0) (3, 1) (4, 0);
    TERM medium: = (4, 0) (7, 1) (10, 0);
    TERM large: = (10, 0) (17, 1) (25, 0);
END_FUZZIFY

// Defuzzify output variable 'portaltype' : {'ls', 'sf_dsl_cable_sat', 'mf_dsl', 'mf_cable_sat', 'lf_dsl', 'lf_cable', 'lf_sat'}
DEFUZZIFY portaltype
    TERM ls: = (0, 0) (10, 1) (20, 0);
    TERM sf_dsl_cable_sat: = (10, 0) (20, 1) (30, 0);
    TERM mf_dsl: = (20, 0) (30, 1) (40, 0);
    TERM mf_cable_sat: = (30, 0) (40, 1) (50, 0);
    TERM lf_dsl: = (40, 0) (50, 1) (60, 0);
    TERM lf_cable: = (50, 0) (60, 1) (70, 0);
    TERM lf_sat: = (60, 0) (70, 1) (80, 0);
    METHOD: COG; // Use 'Center Of Gravity' defuzzification method
    DEFAULT: = 0;
END_DEFUZZIFY

RULEBLOCK No1
    RULE 1: IF connectiontype IS dialup OR connectiontype IS isdn THEN portaltype IS ls;
    RULE 2: IF farmscale IS marginal AND (connectiontype IS dsl OR connectiontype IS cable) THEN portaltype IS ls;
    RULE 3: IF farmscale IS marginal AND connectiontype IS satellite THEN portaltype IS sf_dsl_cable_sat;
    RULE 4: IF farmscale IS small AND (connectiontype IS dsl OR connectiontype IS cable OR connectiontype IS satellite) THEN portaltype IS sf_dsl_cable_sat;
    RULE 5: IF farmscale IS medium AND connectiontype IS dsl THEN portaltype IS mf_dsl;
    RULE 6: IF farmscale IS medium AND (connectiontype IS cable OR connectiontype IS satellite) THEN portaltype IS mf_cable_sat;
    RULE 7: IF farmscale IS large AND connectiontype IS cable THEN portaltype IS lf_cable;
    RULE 8: IF farmscale IS large AND connectiontype IS satellite THEN portaltype IS lf_sat;
END_RULEBLOCK

END_FUNCTION_BLOCK
When the Fuzzy program runs it produces the output like below. The output value it generates will be stored in user profile so it can be used in the application to customize the output for users.

The initial evaluation is done when the user is creating their profile. As the user cruises through the application with slight modifications or addition of more rule blocks the ratings in the profile can be updated to meet changing user requests and tastes. In the sample application only one rule block with seven rules are defined but as the real world complexity increases they can be easily incorporated into the fuzzy systems with more rules or rule blocks in fcl file.

<table>
<thead>
<tr>
<th>Rule ID</th>
<th>Rule</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 (0.0) if (connectiontype IS dialup) OR (connectiontype IS isdn) then portaltype IS ls [weight: 1.0]</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2 (0.0) if (farmscale IS marginal) AND ((connectiontype IS dsl) OR (connectiontype IS cable)) then portaltype IS ls [weight: 1.0]</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3 (0.0) if (farmscale IS marginal) AND (connectiontype IS satellite) then portaltype IS sf_dsl_cable_sat [weight: 1.0]</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4 (0.0) if (farmscale IS small) AND ((connectiontype IS dsl) OR (connectiontype IS cable)) OR (connectiontype IS satellite) then portaltype IS sf_dsl_cable_sat [weight: 1.0]</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>5 (0.5) if (farmscale IS medium) AND (connectiontype IS dsl) then portaltype IS mf_dsl [weight: 1.0]</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>6 (0.0) if (farmscale IS medium) AND (connectiontype IS cable) OR (connectiontype IS satellite) then portaltype IS mf_cable_sat [weight: 1.0]</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>7 (0.0) if (farmscale IS large) AND (connectiontype IS dsl) then portaltype IS lf_dsl [weight: 1.0]</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>8 (0.0) if (farmscale IS large) AND (connectiontype IS cable) then portaltype IS lf_cable [weight: 1.0]</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>9 (0.0) if (farmscale IS large) AND (connectiontype IS satellite) then portaltype IS lf_sat [weight: 1.0]</td>
<td></td>
</tr>
</tbody>
</table>

Output value: 35.500011520117944

5. Conclusion and Future Work

This paper proposes how fuzzy logic can be integrated into an application to handle uncertainty and vagueness. This gives a huge power to transform the imprecision in user’s request to defuzzled values which can be used more easily in applications. Fuzzy logic especially the usage of fuzzy linguistic variables helps add human like reasoning mechanism. This is this enhanced ability to query data more crisply produces the only required results to users.

This paper shows the application of fuzzy logic to a specific domain but with slight modifications it can be extended to other domains also. There are two user input variables which are considered as fuzzy input. More variables can be considered to be added as inputs and the logic of finding the crisp output can be enhanced based on output and the feedback of users.

6. References