

Gathering information on the Web using fuzzy linguistic agents and Semantic Web technologies

Enrique Herrera-Viedma
Dpt. of Computer Science and AI,
University of Granada. Granada,
Spain
viedma@decsai.ugr.es

Eduardo Peis
Dpt. of Library and Information
Science, University of Granada.
Granada, Spain
epeis@ugr.es

José M. Morales-del-Castillo
Dpt. of Library and Information
Science, University of Granada.
Granada, Spain
josemdc@ugr.es

Abstract

In this paper we define a model of a web multi-agent system that combines the use of Semantic Web technologies together with the application of user profiles to carry out information retrieval and filtering tasks. It is designed using fuzzy linguistic techniques which allow dealing with information in a user-friendly way. The system activity is developed in two phases: a retrieval phase to gather the documents from the Web and a feedback phase to update the elements used to filter the documents. We focus on the analysis of the feedback phase. With this multi-agent system model the filtering possibilities are increased and, consequently, also the retrieval capabilities on the Web.

Keywords: Information retrieval, Intelligent agents, Fuzzy linguistic modeling, Semantic Web technologies, User profiles, Filtering.

1 Introduction

The Internet harbours a vast amount of information that does not cease to grow exponentially. To avoid arriving to a situation of collapse, the actual model of Web needs to incorporate tools capable to handle and manage efficiently this huge quantity of resources. The solutions that exist nowadays are shown little efficient since they oblige users to lose great part of their time in deciding which documents are relevant and which not. Therefore it is becoming necessary to develop systems for searching and

mining the Web that permit to improve the access to the information in an efficient way. At this moment, some of the more recurrent technologies to face this problem deal with the development of intelligent software agents [16], the application of techniques of information filtering [23], and the development of the Semantic Web project [3].

Software agents applied to a Web-based environment are organized in distributed architectures [4, 6, 15, 17] to mainly perform tasks of intermediation between users and the Web, for example, assisting users in the information retrieval process [4, 15, 24]. These agents are entities capable to act in an autonomous way, processing and exchanging results with other agents [11, 16].

Nevertheless the main problem of using agents is to find a flexible and agile communication protocol for exchanging information among agents, and between users and agents because of the great variety of forms the information is represented in the Web. Applying fuzzy linguistic techniques we could reduce these communication problems by means of the use of linguistic labels [25], solving thus the problem of quantifying qualitative concepts.

On the other hand, information filtering techniques ease users the task of sorting out relevant documents that fit to users' needs, requirements and preferences (mostly defined in a normalized way in the form of user profiles). The efficacy of these profiles depends on the up-to-dateness of the information contained, so there should exist a mechanism capable to dynamically bring up to date this information and to reflect in "real time" the variations on users' behaviour in their interaction with the system.

Another possibility to improve the activity of a multi-agent system could be the use of some of the technologies of the Semantic Web project [3] that can be exploited to develop ontology-based infrastructures [5, 18, 21] where agents can operate at semantic level with resources described using languages as RDF (Resource Description Framework) in a manner both interpretable by humans and machines.

The aim of this paper is to present a new model of fuzzy linguistic multi-agent system that involves the use of the Semantic Web technologies and user profiles dynamically updated using linguistic matching functions to overcome the problems of the system presented in [14], specifically focusing on its feedback phase.

The paper is structured as follows. Section 2 reviews the methodological tools employed in this research: the fuzzy linguistic tools, the filtering tools, and the Semantic Web technologies. Section 3 presents the new multi-agent model. Finally, some concluding remarks are pointed out in section 4.

2 Methodological Tools

In this section, we present the tools that we apply to design our fuzzy linguistic multi-agent model.

2.1 Fuzzy linguistic tools

The *fuzzy linguistic approach* [25] and in particular the *ordinal fuzzy linguistic approach* [13] are approximate techniques appropriate to deal with qualitative aspects of problems. An ordinal fuzzy linguistic approach is defined by considering a finite and totally ordered label set in the usual sense

$$S = \{s_i, i \in H = \{0, \dots, T\}\}$$

and with odd cardinality (7 or 9 labels). The mid term represents an assessment of "approximately 0.5" and the rest of the terms are placed symmetrically around it. The semantics of the linguistic term set is established from the ordered structure of the term set by considering that each linguistic term for the pair (s_i, s_{T-i}) is equally informative. For each label s_i is given a fuzzy number defined on the [0,1] interval, which is described by a linear trapezoidal membership function represented by the 4-tuple $(a_i, b_i, \alpha_i, \beta_i)$ (the first two parameters indicate the interval in which the membership value is 1.0; the third and fourth

parameters indicate the left and right widths of the distribution). Furthermore, we require the following properties:

1. – *The set is ordered*: $s_i \geq s_j$ if $i \geq j$.
2. – *Negation operator*: $Neg(s_i) = s_j$, with $j = T - i$.
3. – *Maximization operator*: $MAX(s_i, s_j) = s_i$ if $s_i \geq s_j$.
4. – *Minimization operator*: $MIN(s_i, s_j) = s_i$ if $s_i \leq s_j$.

Additionally, we need aggregation operators to combine the linguistic information. In [12] there is defined the Linguistic Ordered Weighted Averaging (LOWA) operator, which has been satisfactorily applied in different fields [13, 14].

2.2 Filtering techniques

Information filtering techniques deal with a variety of processes involving the delivery of information to people who need it. Operating in textual domains, *filtering systems* or *recommender systems* evaluate and filter the resources available on the Web (usually, HTML or XML documents) to assist people in their search processes, in most cases through the use of filtering agents [20]. Traditionally, these systems have fallen into two main categories [19]: *content-based* and *collaborative filtering systems*. In the first category filtering and recommendation are made by matching user query terms with the index terms used in the representation of documents, exploiting the similarity among new documents with already assessed ones, and ignoring data from other users. These recommender systems tend to fail when little is known about user information needs. User profiles solve this problem characterising users through explicit and implicit inputs that define both personal and professional information. *Collaborative filtering systems* use these inputs from many users to filter and recommend documents to users with similar interests, ignoring the representation of documents. These recommender systems tend to fail when little is known about a user, or when he/she has uncommon interests [19]. Several researchers are exploring hybrid content-based and collaborative recommender systems to smooth out the disadvantages of each one of them [1, 7, 19, 23].

2.3 Semantic Web technologies

The Semantic Web is an extension of the present Web, in which the information is gifted of a well

defined meaning, permitting a better cooperation between humans and machines [3]. Basically it is based on the semantic mark up of resources and the development of “intelligent” software agents capable to operate with these resources at semantic level [11].

The semantic backbone of the model is RDF/XML [2], a vocabulary that provides the necessary means to codify, exchange, link, merge and reuse structured metadata from distributed sources in a manner directly interpretable by machines. RDF/XML structures the information in assertions (resource-property-value triples), and uniquely identifies resources by means of URI's (Universal Resource Identifier), allowing software agents to perform inference reasoning over resources (such as documents, user profiles or even queries) using web ontologies [8, 9].

3 The fuzzy linguistic multi-agent model based on Semantic Web and user profiles

In [14] we defined a model of a fuzzy linguistic multi-agent system to gather information on the Web that presents a hierarchical architecture composed of seven action levels: *internet users*, *interface agent*, *collaborative filtering agent*, *task agent*, *content-based filtering agent*, *information agents* and *information sources*. The main novelty of this model is the introduction of collaborative filtering agents in its architecture in order to increase its information filtering capabilities on the Web. Then, it develops its activity in two phases:

- *Retrieval phase*: Coincides with the information gathering process developed by the multi-agent model itself, i.e., this phase begins when a user specifies his/her query and finishes when he/she chooses his/her desired documents among the relevant documents retrieved by the system.
- *Feedback phase*: Coincides with the updating process of collaborative recommendations on desired documents existing in a collaborative recommender system, i.e., this phase begins when the interface agent informs the documents chosen by the user to the collaborative filtering agent and finishes when the recommender system recalculates and updates the recommendations of the desired documents.

The main limitation of this model is that it does not utilise user profiles to characterise user's preferences and this limits its performance possibilities.

To overcome the limitations of the model presented in [14] we define a new and enhanced model of fuzzy linguistic multi-agent system that improves information retrieval by means of the application Semantic Web technologies to set a base for the operation of software agents, and user profiles to enrich the filtering activity.

This model presents a hierarchical structure with six action levels (*internet users*, *interface agent*, *filtering agent*, *task agent*, *information access*, and *information bases*), also two main activity phases, and a set of agents (*interface agent*, *filtering agent*, *task agent*, *profile agent*, *recommendation agent*, *information agents*) that work depending on the phase activity (see Fig. 1):

- **Semantic Retrieval Phase**: This phase is similar to that developed by the model presented in [14]. But there are two main novelties in this phase. On the one hand the use of semantic query languages [10, 22] instead of Boolean ones, due to their capacity for comparing both literal and semantics structures. On the other hand, the underlying semantic infrastructure set by web ontologies that allows, for example, the development of tools to improve the retrieval capacities of the system (such as, for example, a thesaurus). In this phase the following entities participate: *internet users* (level 1), *interface agent* (level 2), *filtering agent* (level 3), *task agent* (level 4), *information agent* as entity of information access (level 5), and *information sources* as information bases that store documents (level 6).
- **Feedback Phase**: Coincides with the updating process of both user profiles and collaborative recommendations on desired documents. Therefore, it involves two processes that need users' appraisals: “profile updating” process and “recommendation” process, where users are given to assess, respectively, the quality of the global answer provided by the system and the retrieved documents. The “profile updating” process consists on the dynamic updating of the user profiles on the basis of the satisfaction degree the user expresses with respect to the global results provided by system. In this process participate the following entities: *internet users* (level 1), *interface agent* (level 2), *profile agent* as entity of information access (level 5), and the *user profiles repository* as information bases that store users' preferences (level 6). The recommendation process is similar to that defined in [14], allowing

users to express their opinions about any retrieved document. Then, the system can use these opinions to recalculate and update the stored recommendations about desired documents. Due to the controlled domain of the system these recommendations are intended to be useful to any user, independently of their profiles (using stereotypes will be one of our future research lines). In this process the entities that participate are the following: *internet users (level 1)*, *interface agent (level 2)*, *recommendation agent as entity of information access (level 5)*, and the *recommendation files repository as information bases (level 6)*.

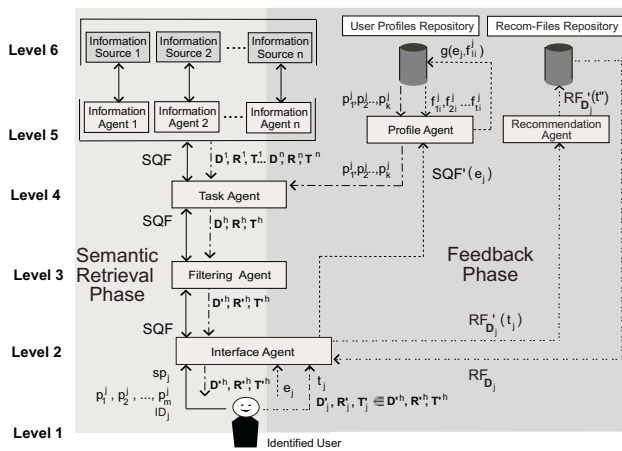


Figure 1: Model architecture with its levels of action and processes.

3.1 Feedback phase: User profile updating process

Before explaining this process we have to clarify the inputs it works on:

- *Semantic Query File (SQF)*: This file is generated when the user formulates its requirements through the interface agent. This agent stores the user's ID_j (defined through the URI that refers to his/her profile), the search parameters (*sp_j*) that define the "semantic" query and a set of preferences {*p₁^j, p₂^j, ..., p_m^j*} on a file in RDF format we call *Semantic Query File (SQF)* (see Fig. 2). These preferences for the user *j* are {*p₁^j, p₂^j, ..., p_k^j*} $0 \leq k \leq m$, being *m* the number of properties used to define a user profile and $p_i^j \in F_i$, being *F_i* the expression domain associated to de property *i*. For example, the user *j* could provide his/her preferences about any of these four categories (*m=4*) of basic preferences: *Document Type, Search aim, Date* or

Search Context. Focusing on the *Search Context* preference, if the domain of work of our system is, for example, "information systems" we could define a expression domain $F_4 = \{decision\ support\ systems, information\ retrieval\ systems, geographical\ information\ systems, data\ mining, knowledge\ representation\}$.

Then, in this case, a user could provide from none (*k=0*) up to four (*k=4*) possible values of preference. Usually he/she will give us just one or two, being the other values taken from his/her stored profile.

```

<Query rdf:ID="query384">
  <user_ID>
    http://www.ugr.es/~user/022005
  </user_ID>
  <preferences_e>
    <preferences rdf:ID="pref_384">
      <docType>SciArticle</docType>
      <context>User_Modeling</context>
      <aim>Research_Article</aim>
      <date>3months</date>
      <value>NULL</value>
    </preferences>
  </preferences_e>
  <search_parameters_e>
    ...
  </search_parameters_e>
</Query>
    
```

Figure 2: Semantic Query File (SQF)

- *User profiles*: We assume a repository storing a set of user profiles determined by the user's ID and characterized by the particular values that each user has assigned to the categories of basic preferences. Each possible preference of the user profile has associated a linguistic frequency property (tagged as <freq>) representing how often a specific value is used in the queries formulated by a user. Thus, if $F_i = \{d_{1i}, d_{2i}, \dots, d_{ti}\}$ then we define $f_i^j = \{f_{1i}^j, f_{2i}^j, \dots, f_{ti}^j\}$ as the set of frequency values associated with each possible value $l \in \{1, \dots, t\}$ of the property *i* in the profile of the user *j* (see Fig. 3). The range of possible values for the frequencies is defined in a set of

seven linguistic labels, $S=\{always, almostAlways, mostTimes, sometimes, aFewTimes, almostNever, never\}$, i.e., $f^j_{li} \in S$.

```
<DocType rdf:ID="docType-pr001">
  <type1_e>
    <Type1 rdf:ID="type1-pr001">
      <type>SciArticle</type>
      <freq>AlmostAlways</freq>
    </Type1>
  </type1_e>
  ...
</DocType>
```

Figure 3: Preferences in a user profile

- *Global satisfaction degree:* When the user receives the documents retrieved by the system he/she is required to provide a linguistic global satisfaction degree e_j that can also be defined using a set of seven linguistic labels $S' = \{Total, veryHigh, high, medium, low, veryLow, Null\}$, i.e., $e_j \in S'$. The satisfaction degree shows if the user is more or less satisfied with the general performance of the multi-agent system in relation with a specific query. We use this information to update the user profile through a simple tacit preferences elicitation mechanism developed to allow the system finding out preferences that are more likely to return satisfactory results to the user.

Assuming the above inputs the profile updating process is developed in the following steps (see Fig. 1):

- **Step 1:** Once the user has checked the set of resources retrieved (D^h, R^h), the interface agent asks the user to express his/her satisfaction degree ($e_j \in S'$) with respect to the overall performance of the system, and it is stored in a SQF.
- **Step 2:** This SQF is transferred from the interface agent to the profile agent.
- **Step 3:** The profile agent carries out the updating of the linguistic frequency of use $\{f^j_{1i}, f^j_{2i}, \dots, f^j_{ii}\}$ of each preference i defined in the user profile. To do that, we propose to use some type of matching function similar to those defined in the information retrieval systems to model the weighted queries. In particular, we use linguistic matching functions to model threshold weights in

weighted user queries. Assuming the linguistic satisfaction degree $e_j \in S'$ expressed by a user, if the property i is assigned a value l , then its respective associated frequency $f^j_{li} \in S$ is updated by means of the following linguistic matching function $g: S' \times S \rightarrow S$:

$$g(e_j, f^j_{li}) = \begin{cases} S_{Min(a+\beta, T)} & \text{if } s_a \leq s_b \\ S_{Max(0, a-\beta)} & \text{if } s_b < s_a \end{cases}$$

such that, (i) $s_a = f^j_{li}$; (ii) $s_b = e_j$; and (iii) β is a bonus value that rewards/penalizes the frequencies of the preferences of the user profile, which can be defined depending on the closeness between f^j_{li} and e_j for example, $\beta = round(2|b-a|/T)$. We should point out that this function is a non-decreasing matching function as the traditional threshold matching functions.

In such a way, when a user selects just some (or even none) preferences in the search interface the system will be able to define on behalf of the user the set of preference values that has obtained better satisfaction results over time (i.e. the system brings to light the implicit preferences of the user, which may not coincide with those used more often).

3.2 Feedback phase: Recommendation process

This process needs two basic inputs:

- *Recommendation Files:* Each document accessed in some moment of its history in the system has an associated recommendation file (RF) in RDF format (see Fig. 4) where is contained information about all the appraisals made by users that have read it formerly. In a RF appears the ID of the document it refers to (defined through its URI), the current recommendation value and a set of log items containing previous appraisals about that document. Each log item is defined by an user's ID, his/her corresponding appraisal and the search context used in the query formulated by the user to retrieve that document (see "Search Context" preference in page 4). This representation enables the adoption of different recommendation updating policies (e.g., topic-based or accredited users' appraisals updating). The decision to separate the documents and their recommendations is justified on the dynamic nature of the recommendations. In such a way, we can modify the RF as many times as we need

leaving untouched the representation of its associated document.

```

<RecomFile rdf:ID="recomf001">
  <doc_ref>
    http://www.ugr.es/~doc/pr008
  </doc_ref>
  <recom_value>High</recom_value>
  <recom_history>
    <R_history rdf:ID="histf001">
      <item>
        <RecomItem rdf:ID="form01-pr001">
          <appraisal>VeryHigh</appraisal>
          <topic>Data mining</topic>
          <user_ID>
            http://www.ugr.es/~user/022005
          </user_ID>
        </RecomItem>
      </item>
      ...
    </R_history>
  </recom_history>
</RecomFile>

```

Figure 4: Representation of a Recommendation File

- *Recommendation value*: When a user checks out a document the systems asks him/her to recommend it to the rest of users of the system according to his/her opinion about the resource, helping in this way other users in the system to decide which resources are worthy to be read. The recommendation value (t_j) provided by user j can be defined using a set of five linguistic labels $S'' = \{veryHigh, high, medium, low, veryLow\}$, i.e. $t_j \in S''$.

Then, this process is carried out in three steps (see Fig. 1):

- **Step 1:** Any document $D'_j \in (D^h, R^h, T^h)$ has an associated relevance degree R'_j and an associated set of “historical” recommendation values $T'_j = \{t'_{1j}, t'_{2j}, \dots, t'_{mj}\}$. When the user checks out the document D'_j is asked by the interface agent to appraise its global quality using a linguistic label $t_j \in S''$. This appraisal, the user’s ID and the “Search context” preference value in the SQF of

the current query (used to retrieve D'_j) are added to a new “*historic search*” item in the RF associated to D'_j ($RF_{D'_j}$). The modified $RF_{D'_j}$ will be called $RF'_{D'_j}$ since now.

- **Step 2:** The $RF'_{D'_j}$ is transferred from the interface agent to the recommendation agent.
- **Step 3:** The recommendation agent recalculates the recommendation value of the document by means of an aggregation function that combines the appraisal t_j with the set of associated “historical” recommendation values T'_j according to the updating policies of the system. To do this, we could use the LOWA operator aforementioned. The resulting recommendation value (t'') is stored in the `<recom_value>` tag of the $RF'_{D'_j}$ (Fig. 4).

One of the assets of this method is its flexibility, since it allows us to set different recommendation policies at low cost, defining their constraints in an ontology that agents can reason over.

4 Concluding remarks

We have described the architecture and elements of a fuzzy linguistic multi-agent system designed to perform information retrieval and filtering tasks in domain dependant environments using jointly Semantic Web technologies and user profiles to provide the system with the necessary infrastructure to improve inference and communication capacities of agents, to represent the information with a common vocabulary both human and machine interpretable and to better characterize users in a way the performance of the system can be increased. We have also proposed a dynamic updating method for user profiles that allows their adaptation to the changes observed in users’ preferences by mean of fuzzy linguistic matching functions.

5 References

- [1] C. Basu, H. Hirsh, W. Cohen, “Recommendation as classification: Using social and content-based information in recommendation”. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence* (1998), pp. 714-720
- [2] D. Beckett, “RDF/XML Syntax Specification (Revised)” (2004). Available at <http://www.w3.org/TR/rdf-syntax-grammar/>. Accessed 02/19/2005

- [3] T. Berners-Lee, J. Hendler, O. Lassila, "The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities", *Sci Am* May (2001)
- [4] W. Brenner, R. Zarnekow, H. Witting, "Intelligent Software Agent, Foundations and Applications", (Springer-Verlag. Heidelberg, 1998)
- [5] Esperanto Project. Available at <http://esperanto.semanticweb.org>. Accessed 02/21/2005
- [6] B. Fazlollahi, R.M. Vahidov, R.A. Aliev, "Multi-agent distributed intelligent system based on fuzzy decision making". *Int J Intell Syst* 15 (2000), pp. 849-858
- [7] N. Good, J.B. Shafer, J.A. Konstan, A. Borchers, B.M. Sarwar, J.L. Herlocker, J. Riedl, "Combining collaborative filtering with personal agents for better recommendations". In *Proceedings of the Sixteenth National Conference on Artificial Intelligence* (1999), pp. 439-446
- [8] T.R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing". In *Int J Hum-Comput St* 43 (5-6) (1995), pp. 907-928
- [9] N. Guarino, "Formal ontology and information systems". In N. Guarino (ed) *Formal Ontology in Information Systems, Proceedings of FOIS'98*. Trento (Italy), (IOS Press, Amsterdam, 1998), pp. 3-17
- [10] R. Guha, R. McCool, E. Miller, "Semantic search". In *12th International World Wide Web Conference, WWW2003*, Budapest (Hungary), (2003) pp. 700 - 709
- [11] J. Hendler, "Agents and the Semantic Web", *IEEE Intel Syst*, March, April (2001), pp. 30-37
- [12] F. Herrera, E. Herrera-Viedma, J.L. Verdegay, "Direct Approach Processes in Group Decision Making using Linguistic OWA operators", *Fuzzy Set Syst* 79 (1996), pp. 175-190
- [13] E. Herrera-Viedma, E. Peis, "Evaluating the informative quality of documents in SGML format using fuzzy linguistic techniques based on computing with words", *Inform Process Manag* 39 (2) (2003), pp. 195-213
- [14] E. Herrera-Viedma, F. Herrera, L. Martínez, J.C. Herrera, A.G. López, "Incorporating Filtering Techniques in a Fuzzy Multi-Agent Model for Gathering of Information on the Web", *Fuzzy Set Syst* 148 (1) (2004), pp. 61-83
- [15] N. Jennings, K. Sycara, M. Wooldridge, "A roadmap of agent research and development", *Autonomous Agents and Multi-Agents Systems* 1 (1998), pp. 7-38
- [16] P. Maes, "Agents that reduce the work and information overload", *Commun ACM*, 37 (7) (1994), pp. 30-40
- [17] A. Moukas, G. Zacharia, P. Maes, "Amalthea and Histos: Multiagent systems for WWW sites and representation recommendations". In M. Klusch (ed), *Intelligent Information Agents* (Springer-Verlag, 1999), pp. 293-322
- [18] Ontoknowledge Project. Available at <http://www.ontoknowledge.org/>. Accessed 02/18/2005
- [19] A. Popescul, L.H. Ungar, D.M. Pennock, S. Lawrence, "Probabilistic models for unified-collaborative and content-based recommendation in sparse-data environments", In *Proc of the 17th Conference on Uncertainty in Artificial Intelligence (UAI)*, San Francisco (2001), pp. 437-444
- [20] J.B. Schafer, J.A. Konstan, J. Riedl, "E-Commerce recommendation applications". *Data Min Knowl Disc* 5 (1/2) (2001), pp. 115-153
- [21] Semantic Web Advanced Development for Europe (SWAD-Europe). Available at <http://www.w3.org/2001/sw/Europe/>. Accessed 02/17/2005
- [22] U. Shah, T. Finin, Y. Peng, J. Mayfield, "Information Retrieval on the Semantic Web", In *Proc of the 10th Int Conference on Information and Knowledge Management* (2002), pp. 461-468
- [23] B. Shapira, U. Hanani, A. Raveh, P. Shoval, "Information filtering: A new two-phase model using stereotypic user profiling", *J Intell Inform Syst* 8 (1997), pp. 155-165
- [24] R.R. Yager, "Intelligent agents for World Wide Web advertising decisions", *Int J Intell Syst* 12 (1997), pp. 379-390
- [25] L.A. Zadeh, "The concept of a linguistic variable and its applications to approximate reasoning". Part I, In *Inform Sciences* 8 (1975), pp. 199-249. Part II, In *Inform Sciences* 8 (1975), pp. 301-357. Part III, In *Inform Sciences* 9 (1975), pp. 43-80