We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Information Extraction from Text – Dealing with Imprecise Data

Turksen, I.Burhan[1,2] and Celikyilmaz, Asli[3]
[1]Department of Industrial Engineering TOBB-Economics and Technology University,
[2]University of Toronto,
[3]Computer Sciences Department University of California, Berkeley,
[1]Turkey
[2]Canada
[3]USA

1. Introduction

Information Extraction from text is a special case of Data Mining where one extracts valuable information from unstructured documents. On the other hand, soft computing approaches, e.g., neural networks, fuzzy systems, deal with information processing. An architecture that can combine these processes into a complete system has been the top research field in computer and information sciences for the last decade. In this paper we will present novel methods for information processing, which can model imprecision in a given database that classical bivalent methods cannot handle. Specifically we will present novel approaches on developing soft models via function representations in place of rule based methods. We will present examples on more intelligent applications of information extraction from text and compare the performance of the novel approaches to the state-of-the-art learning methods on this field.

There have been vast amount of work on information processing, which keeps us listing them all in here. Since the aim of this chapter is to present novel approaches on information processing via fuzzy functions and their extensions, we will start with the related work on functional analysis on information processing. Later in section 3, we introduce the framework of fuzzy system modelling with fuzzy functions followed by extensions of fuzzy functions under uncertainties in section 4. Specifically, we present various fuzzy system modelling approaches via higher order fuzzy sets, e.g., interval-valued type-2 and full type-2 fuzzy modelling. Section 5 presents possible applications of the latter novel approaches on information extraction from text. In section 6 we present the results of this study and discussions for future research. Finally, in section 7 we draw conclusions.

2. Related wok on information processing with functional representations

Let us first briefly review the literature to expose a historical account of "fuzzy function?" in a variety of approaches by several authors.

Originally, "Fuzzy Functions" were defined in (Bandler & Grinder, 1976) as a connecting or overlapping of our sensory representational systems. Technically, Bandler and Grinder define "fuzzy functions" as:

"...Any modeling involving a representational system and either an input channel or an output channel in which the input or output channel involved is a different modality from the representational system with which it is being used. In traditional psychophysics, this term, 'fuzzy function', is most closely translated by the term 'synesthesia'..."

Later we find certain articles in the literature, for example, (Sasaki, 1993) and (Demirci, 1999), etc…

Turksen (2006) first introduced "Fuzzy Functions" unaware of the publications stated above and published "Fuzzy Functions with LSE" (Turksen, 2008) which is quite different in structure and intent from Sasaki and Demirci expositions. Later "Fuzzy Functions" were further developed in a variety of directions in (Celikyilmaz & Turksen, 2007; 2008a-g; 2009a-d; Turksen & Celikyilmaz, 2006).

With this perspective, Fuzzy Functions, for short, FF, are proposed for the structure identification of system models and reasoning with them. These fuzzy functions can be determined by any function identification method such as least squares' estimates, LSE, maximum likelihood estimates, MLE, support vector machine estimates, SVM (Gunn, 1998) etc. Furthermore, our work extends to Type 2 Fuzzy Functions which incorporates the parameter uncertainties in system modelling.

3. Building fuzzy system models with fuzzy functions

3.1 Background of fuzzy rule bases
Traditional FIS structure is based on the fuzzy rule base (FRB) (*if-then* rules) structures,

$$R_i: IF\ antecedent_i\ THEN\ consequent_i \qquad (1)$$

In (1) each R_i, $i=1…c$, represents one fuzzy rule. Based on the representation of the consequents structure, FISs get the name; *Linguistic* FIS when the consequents are represented with fuzzy sets as in (Zadeh, 1965), *Mizumoto* FIS (Mizumoto, 1989) when the consequents are represented with a scalar value, *Takagi-Sugeno FRB* (Takagi & Sugeno, 1985) when the consequents are represented with linear or non-linear equations of input variables. For illustration, *Takagi-Sugeno* FIS structure is defined as;

$$R_i: IF\ AND_i^{NV}\ (x_j \in X_j\ is\ A_{ij})\ THEN\ y_i = a_i x^T + b_i \qquad (2)$$

In (2) A_{ij} is the type-1 fuzzy set characterized by a type-1 membership function, $\mu_A(x_j) \rightarrow [0,1]$, where $x_j \in X_j$ is the jth input variable. $a_i = (a_{i,1} … a_{i,NV})$ and b_i are regression coefficients of ith rule. A type-1 fuzzy set is identified for each input variable, assuming they are independent from one another, viz. non-interactivity assumption. Fuzzy connectives such as t-norm are used to combine antecedent fuzzy sets to calculate the degree of fire of each rule.

The traditional FIS structures presented above have various challenges that should not be neglected (Turksen & Celikyilmaz, 2006). Among some of these challenges are identification of the; types of antecedent and consequent membership functions, and their varying parameters, most suitable combination operators (t-norm, t-conorm, etc.), conjunction operators during aggregation of antecedents, and consequents, implication operator types to capture uncertainty associated with the linguistic "AND", "OR", "IMP" for the representation of the rules, and reasoning with them, type of defuzzification method, etc.

The literature indicates that a given FIS model performance can be slightly affected by the change in t-norm values. Nevertheless, one still needs to decide the type of t-norm and t-conorm operators. Over the course of many years these challenges have been investigated to reduce the fuzzy operations (Babuska & Verbruggen, 1997), and expert intervention and many different methods are proposed such as building hybrid fuzzy systems using other soft computing methods via genetic algorithms or neural networks, etc.

Some extensions of traditional FISs e.g., (Uncu et.al., 2004), assume that antecedent fuzzy sets are dependent on each other (interactive), so in these systems an entire antecedent part of a given rule is represented with a single type-1 fuzzy set. Such FIS structures are expressed as follows:

$$R_i: IF\ x \in X\ is\ A_i\ THEN\ y_i = a_i x^T + b_i \qquad (3)$$

In (3) the fuzzy set A_i is characterized by a type-1 membership function $\mu_i(x) \rightarrow [0,1]$ where $x \in X$ is an input vector.

Later, the performance of latter systems is improved with the implementation of improved fuzzy functions algorithm (Celikyilmaz & Turksen, 2008a-g). Next subsection briefly reviews such systems, which forms the basis of the Type-2 Fuzzy Functions.

3.2 Enhanced FIS with improved fuzzy functions

Although FSM approach based on Fuzzy Functions in Fig. 1 and traditional FSM approaches based on FRB structures (Takagi & Sugeno, 1985; Emami et.al. 1998; Bodur et al., etc.) share similar system design steps, they differ in structure identification, namely in finding the fuzzy models (rules) for each pattern identified. The new FFs approach first clusters a given data into several overlapping fuzzy clusters, each of which is used to define a separate decision rule. Fuzzy c-means clustering (FCM) (Bezdek, 1984) has been the main clustering algorithm utilized in these methods to find fuzzy partitions so far. The novelty of the FFs approaches are that, during structure identification, similarity of the objects are enhanced with additional fuzzy identifiers viz. membership values, by utilizing them as additional predictors of the system model along with the original input variables to estimate the local relations of the input-output data. Thus, membership values and their list of possible (user-defined) transformations are augmented to original dataset as new dimensions to structure different representations for each cluster.
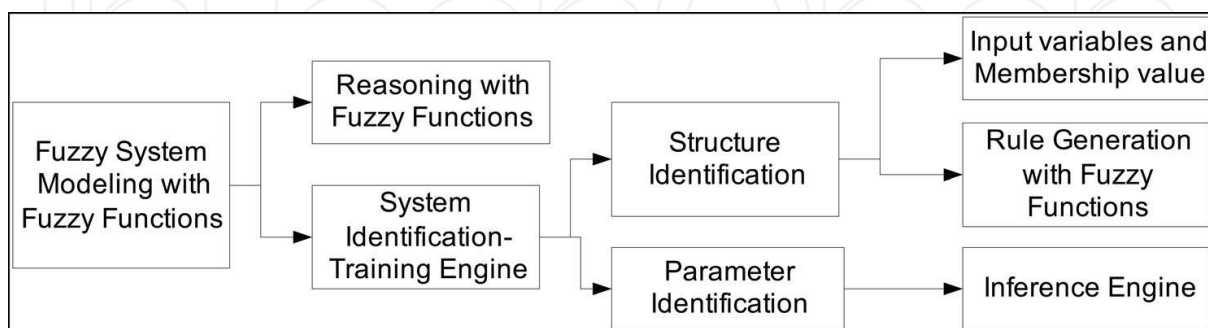

Fig. 1. Framework of fuzzy system models with fuzzy functions approach.

In (Celikyilmaz & Turksen, 2008b) a new fuzzy clustering algorithm is proposed, namely Improved Fuzzy Clustering (IFC) algorithm, which carries out two objectives: (i) to find good representation of the partition matrix, which captures the multiple model structure of

the given system by identifying the hidden patterns, (ii) to find the membership values, which are good predictors of the regression models of each cluster. Therefore the objective function of the new IFC is designed based on these two objectives. The novelty of the presented fuzzy clustering approach, which aparts itself from the earlier improved fuzzy clustering approaches by (Chen et al. 1998; Höppner & Klawonn, 2003; Menard, 2001) is that, during IFC optimization, regression models, to be build for each cluster, will use only membership values measured at a particular iteration and their user defined transformations, but not the original input variables. Alienating original input variables and building regression models with membership values will shape the memberships into candidate inputs to explain the output variable for each local model. As a result of this improvement, the new IFC introduces a new membership function. In the proposed IFC, we hypothesize to find membership values that can increase the prediction power of the system modeling with FFs. In this sense, the resulting fuzzy functions are referred as "improved fuzzy functions (IFF)".
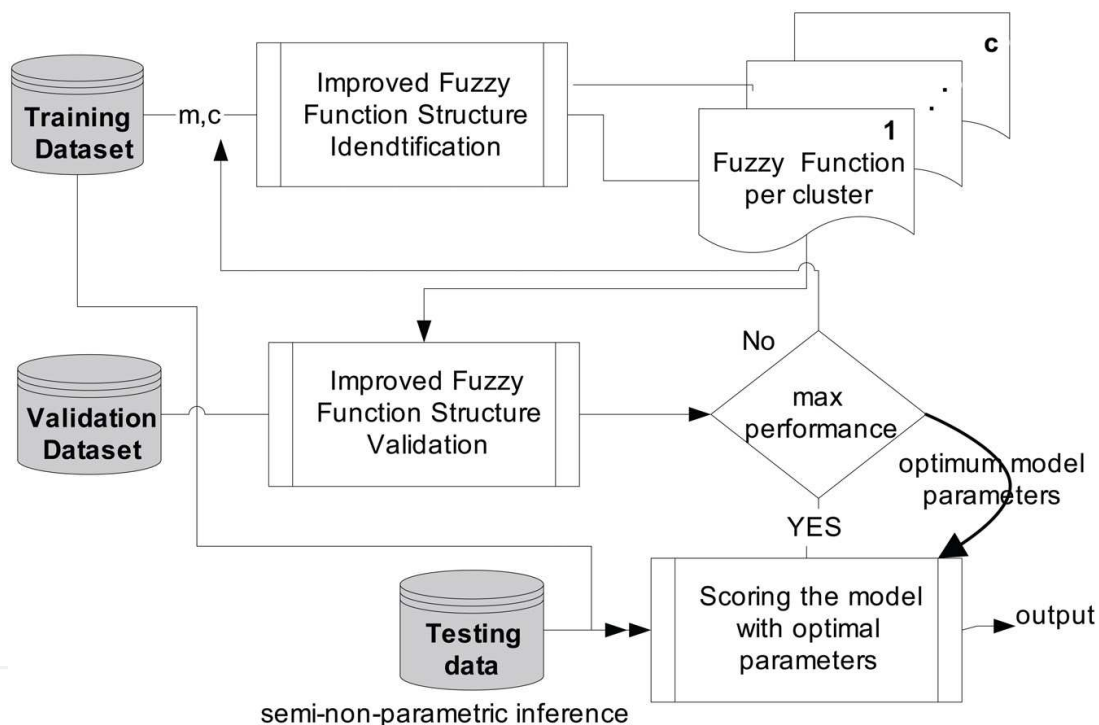

Fig. 2. Flow chart of Fuzzy System Models with Fuzzy Functions Approach.

Structure identification of FIS with Fuzzy Functions Systems is based on Improved Fuzzy Clustering (IFC) algorithm to identify the hidden structures in a given dataset. The learning algorithm is sketched in Fig.2.

The type-1 FIS with Improved Fuzzy Functions (Celikyilmaz & Turksen, 2007, 2008b) is designed to eliminate most of the aforementioned fuzzy operations of traditional type-1 FIS. In somewhat simplified view, such fuzzy systems work as follows:

- The domain $X \subseteq \Re^{nv}$ with nv dimensional input space is partitioned into c overlapping clusters using IFC, and each cluster is represented with cluster centers, V_i, $i=1,..,c,$ and membership value matrix, U_i.
- To each of these regions a local fuzzy model $f_i: V_i \rightarrow \Re$ is assigned by using membership values as additional predictors to given input vector, $x \in X$. The system then identifies

one fuzzy output from each fuzzy model and then weights these outputs based on the membership values of the given input vector in each cluster.

Let (x_k, y_k) denote each training data point, where $x_k(x_{1,k}...x_{nv,k})$, is the kth input vector of nv dimensions, y_k, is their output value, $\mu_{ik} \in [0,1]$ represent the membership value of kth vector to cluster $i=1...c$, c be the total number of clusters, m, be the level of fuzziness parameter. The learning algorithm of type-1 FIS with the Improved Fuzzy Functions approach (Celikyilmaz & Turksen, 2007; 2008b;c) is processed as follows:

Step 1: IFC is a dual-structure clustering method combining FCM (Bezdek, 1984) and fuzzy c-regression algorithms (Höppner & Klawonn, 2003) within one clustering schema and has the following objective function:

$$\min J_m^{IFC} = \sum_{i=1}^{c} \sum_{k=1}^{n} \mu_{ik}^m d_{ik}^2 + \sum_{i=1}^{c} \sum_{k=1}^{n} \mu_{ik}^m E_{ik} \qquad (4)$$

In (4), $d_{ik}=||x_k-v_i||$, represents the Euclidean distance of each x_k to each cluster center, v_i. The error $E_{ik}=(y_k-g_i(\tau_{ik}))^2$ is the total squared deviation between of the approximated fuzzy models, namely the *interim* fuzzy functions, $g_i(\tau_i)$ of cluster i and the actual output. The novelty of each $g_i(\tau_i)$ is that corresponding membership values and their possible transformations are the only predictors of interim fuzzy functions, while excluding original variables. The aim is to calculate the membership values that can be candidate input variables when used to estimate the local models. An example interim fuzzy function can be formed using:

$$g_i(\tau_i, \hat{w}_i) = \hat{w}_{0i} + \hat{w}_{1i}\mu_i + \hat{w}_{2i}(1 + \exp(-\mu_{ik}^m)) \qquad (5)$$

In (5), $\hat{w}_i$ represents the vector of regression coefficients. IFC minimizes the objective function, J_m^{IFC}. The second term of the objective function can be minimized if optimum functions can be found. Thus, the algorithm searches for the best interim fuzzy functions, $g_i(\tau_i)$.

From the Lagrange transformation of the objective function in (4) the membership values are calculated with a new membership value update equation as follows,

$$\mu_{ik} = \left(\sum_{j=1}^{c} \left[\left(d_{ik}^2 + E_{ik}^2 \right) \Big/ \left(d_{jk}^2 + E_{jk}^2 \right) \right]^{1/(m-1)} \right)^{-1} ; \sum_{i=1}^{c} \mu_{ik} = 1 \qquad (6)$$

, $i=1...c$, $k=1...n$. Punishing the objective function with an additional error, forces to capture the membership values that would help to improve the local models, but at the same time identify the clusters. Thus, the new membership function yields a matrix of "*improved*" membership values, $\mu_{ik}^* \in U^* \subset \Re^{n \times c}$. It has been proven that the improved membership values obtained from the IFC can predict the local relations better than the membership values obtained from the FCM clustering algorithm.

Proposed IFC optimization method searches for optimum membership values, which are to be used later as additional predictors to estimate parameters of Fuzzy Functions of a given system model. The structures of functions to be approximated depend on distribution of membership values with an output variable. One should choose appropriate membership value transformations to approximate output variable. For any given fuzzifier m and number of clusters c the outputs of the IFC algorithm are as follows:

- optimum parameters of fuzzy functions f(τi) of each cluster ŵi, i=1…c, that are captured from the last iteration step,
- structure of the input matrix, τi, viz. the list of different types of membership value transformations that are used to approximate each $f(τi)$ during IFC,
- optimized membership matrix, U*(x,y), the cluster centers v*(x,y)

(*) indicates the optimum results from the new IFC algorithm.

Step 2: One fuzzy function is approximated for each cluster to identify the input-output relations in local model for each cluster *i*. The dataset of each cluster is comprised of the original input variables, *x*, improved membership values of particular cluster *i* obtained from IFC, and their user defined transformations. This is same as mapping the input space, $\Re^{nv}$, of each individual cluster *i* onto a higher dimensional feature space $\Re^{nv+nm}$, i.e., $x \rightarrow \Phi_i(x, \mu_i^*)$, where *nm* is the total number of membership value transformations used to structure a system of *principle fuzzy functions*. Parameters of an optimum regression function are sought in this new space. The principle fuzzy functions, $\hat{f}_i(\Phi_i)$, to determine the local relations of each cluster are structured in (*nv+nm*) space.

The interim fuzzy functions, $g_i(τ_i)$ are different from principle fuzzy functions $\hat{f}_i(\Phi_i)$, since $g_i(τ_i)$ is used only for shaping the membership functions during IFC algorithm and only use membership values and their transformations only as input variables. A prominent feature of the principle fuzzy function approximation of such forms is that, if the relations between input and output variables cannot be defined in the original space, we can use proposed fuzzy functions approach to explain their relationship in the $\Re^{nv+nm}$ space.

Step 3: An approximate optimum number of clusters, *c**, of IFC algorithm is determined with the cluster validity index, *cviFF* (Celikyilmaz & Turksen, 2009a;2008c), designed to evaluate the IFC algorithm with:

$$cviFF = \frac{vc^*}{(c \cdot vs^*)+1} \ , \ \ vc^* = \max\frac{1}{n}\sum_{k=1}^{n}\mu_{ik}^{*m}\left(d_{ik}^{\ 2}+(y_k-\hat{f}_{ik}(\Phi_i))^2\right)$$

$$vs^* = \begin{cases} \min\limits_{i,j\neq i}\left(\left\|v_i-v_j\right\|^2 \big/ \left|\left\langle\alpha_i,\alpha_j\right\rangle\right|\right), & if \left|\left\langle\alpha_i,\alpha_j\right\rangle\right| \neq 0, \\ \min\limits_{i,j\neq i}\left\|v_i-v_j\right\|^2, & otherwise \end{cases} \tag{7}$$

In (7) *vc** represents the *compactness* and *vs** represents the *separability*. *vc** combines within-cluster distances and errors between actual and estimated output obtained from *c* number of principle fuzzy functions. The v_i and v_j *i,j*=1,..,*c, i≠j* represent the cluster center vectors of two separate clusters of an IFC model. *vs** determines the structure of clusters by measuring the ratio of cluster center distances to the angle between their regression functions. The a_i in the $|\langle a_i,a_j\rangle| \in [0,1]$, *i,j=1,…,c,* is the unit normal vector of each principle fuzzy function *i*, $\hat{f}_i(\Phi_i)$, $a_i=[n_i]/||n_i||$. The absolute value of inner product of unit vectors of two fuzzy functions of two different clusters, $|\langle a_i,a_j\rangle| \in [0,1]$, *i,j=1,…,c, i≠j*, equals to the value of *cosine* of the angle between them: $cos\theta_{i,j} = \langle n_i,n_j\rangle/|n_i|*|n_j| = \langle a_i,a_j\rangle$. When two cluster centers are too close to each other due to oversized number of clusters, the distance between them becomes almost ($\cong 0$) invisible, then validity measure goes to infinity. To prevent this, the denominator of *cviFF* in (7) is increased by 1.

Any regression approximation method can be employed to identify the parameters of local functions, e.g. LSE or soft computing approaches such as neural networks or support vector machines (SVM) (Gunn, 1998). For instance, when LSE is used to identify the local models of a cluster *i*, the principle fuzzy function is formed with function as:

$$\hat{y}_i = \hat{f}_i\left(x, \Phi_i\right) = \beta_{0,i} + \beta_{1,i}\mu_i^* + \beta_{2,i}x \tag{8}$$

Step 4: Finally, one crisp output is obtained by taking the average weight of the outputs from each principle function *i*, with corresponding membership values as follows:

$$\hat{y} = \sum_{i=1}^{c^*} \mu_i^* \hat{f}_i\left(\Phi_i\right) \tag{9}$$

The experiments indicate that the FIS system based on Fuzzy Functions (Turksen, 2008; Celikyilmaz & Turksen, 2008a) outperform traditional type-1 FIS as well as other soft computing approaches. One of the issues of this approach is that since type-1 fuzzy sets are implemented, it may not be possible to handle uncertainties. In particular, there is also the uncertainty in determining the system parameters such as; type of membership value transformations (τ_i) used during IFC algorithm (such as in (5)) and during shaping principle fuzzy functions, $\hat{f}_i(\Phi_i)$ (such as in (8)). Hence, we implement interval type-2 fuzzy sets into fuzzy functions system. Using the type-2 FIS instead of type-1 FIS in Fuzzy Function systems has many advantages, which are summarized as follows:

- The type-2 fuzzy sets can handle the numerical uncertainties in inputs and outputs of fuzzy functions,
- The uncertainty in determining the type, and parameters of membership value extraction functions are managed,
- The type-2 fuzzy sets are discretisized into a large number of embedded type-1 fuzzy sets, which enable a wealthy environment to describe the local input-output relations.

The new type-2 FIS based on Fuzzy Functions is designed that can characterize structure of optimum membership value transformations $\Omega=\{\tau_i,\Phi_i\}$ of given fuzzy function, the shape of membership values, the number and type of fuzzy function structures, and number of local structures. In summary, the proposed approach searches for the optimum uncertainty interval of membership functions and optimum list of the fuzzy function structures for each local model using soft computing approaches such as genetic algorithms.

4. Modelling uncertainty with fuzzy functions

4.1 Review of type-2 fuzzy inference systems and variations

Before we present the new type-2 FIS based on Fuzzy Functions, we briefly review the traditional type-2 FISs. For the generalized type-2 case, where the secondary membership functions, the third dimension, are of *any* type, there is a significant computational complexity that has delayed their development (Coupland & John, 2007). Thus, in most type-2 fuzzy logic research, the interval type-2 fuzzy sets are. Nonetheless, recent investigations on full type-2 fuzzy logic systems such as (Coupland & John, 2007) or (Celikyilmaz & Turksen, 2008c) present promising results.

A type-2 fuzzy set $\tilde{A}$ is characterized by a type-2 membership function $\mu_{\tilde{A}}(x,u)$, where $x \in X$ and $u \in J_x \subseteq [0,1]$, i.e.,

$$\tilde{A} = \left\{ \left(x, \left(\mu_{\tilde{A}}, u \right) \right) \mid x \in X, \mu_{\tilde{A}} \in J_x \subseteq [0,1] \right\}. \tag{10}$$

The elements of the domain of $\mu_{\tilde{A}}(x)$ are called the *primary memberships* of x in $\tilde{A}$, and the membership functions of the primary memberships in $\mu_{\tilde{A}}(x)$ are called the *secondary memberships* of x in $\tilde{A}$.

The interval fuzzy logic systems are embedded type-1 fuzzy inference systems, which implement fuzzy sets, $\tilde{A}$. In (10) J_x is a set of real values with finite elements. A special case of interval-valued type-2 FIS is formalized with the fuzzy sets of discrete domain as follows:

$$\tilde{A}^i = \left\{ \left(x, (\mu, 1) \right) \mid x \in X, \mu \in \{ \mu_i \}, \mu_i \in [0,1] \right\} \tag{11}$$

In (11), the membership functions are discretisized and are used to form a collection of embedded type-1 FIS. Hence, ith rule in a type-2 system having nv inputs $x_1 \in X_1 \dots x_{nv} \in X_{nv}$ and one output $y \in Y$ is represented with;

$$\tilde{R}_i : \text{ IF } AND_{j=1}^{nv} \ (x_j \in X \text{ is } \tilde{A}_{ji}) \text{ THEN } y_i \text{ is } \tilde{B}_i \tag{12}$$

The uncertainty in primary membership functions of a type-2 fuzzy set $\tilde{A}$, is represented with a bounded region that is called the *foot-print of uncertainty* (FOU). It is the union of all the primary membership functions. With the implementation of type-2 fuzzy sets, determining the optimum type-1 membership function reduces its significance.

In order to extract crisp output, the type of the set is first reduced with *a type reduction* process, which is an extension of defuzzification method. Then type reduced set is defuzzified to obtain a zero order (crisp) output. The foundations of type-2 fuzzy logic system are explained in (Mendel, 2001) in more detail.

The type-2 fuzzy set parameters associated with each variable in each rule are identified mostly using supervised learning methods. In (Uncu et.al., 2004) the FCM (Bezdek, 1984) clustering is used to identify the hidden structures. They use uncertainty in selection of *level of fuzziness* parameter, m, of FCM as the source of uncertainty of the values of inference parameters and identify embedded type-1 FIS for each m to represent discrete interval type-2 FIS (DIT2FIS). Let m^r be the rth level of fuzziness, $m^r \in \{m^1 .. m^{NM}\}$, where NM is the number of disjoint m values. Thus, they find rth embedded type-1 fuzzy rule for each different m^r. $\mu_A{}^r$ represents the membership values associated with rth embedded type-1 fuzzy set A. Their Tagaki-Sugeno FIS is as follows:

$$\tilde{R}_i^r : \text{ IF } x \in X \text{ is } A_i{}^r \text{ THEN } y_i{}^r = a_i{}^r x^T + b_i{}^r \tag{13}$$

In (13) $r=1 \dots NM$, and $a_i{}^r x^T + b_i{}^r$ are regression coefficients associated with ith rule of rth embedded type-1 fuzzy rule. Thus, the problem of building type-2 FIS in DIT2FIS is reduced to finding traditional embedded type-1 FISs.

Type-2 FIS based on Fuzzy functions (Celikyilmaz & Turksen, 2009c;2008a) is a different approach to uncertainty modeling which extends inference strategy of (Uncu et.al., 2004) by introducing two separate uncertainty parameters, the level of fuzziness and the fuzzy function structures to form interval type-2 fuzzy sets. In the next we will briefly present type-2 fuzzy functions methods.

4.2 Type-2 fuzzy functions
4.2.1 Interval valued type-2 fuzzy functions

The interval Valued Type-2 Fuzzy Functions, IVT2FF in short, evidently differs from the other type-2 FIS of the previous sections in many ways. For instance, instead of the traditional FIS such as Tagaki-Sugeno structures, the algorithm is based on the Fuzzy Functions structures (Turksen, 2008), which do not require fuzzy connectives (aggregation, implication, defuzzification) and introduce a new fuzzy clustering algorithm. In addition, the uncertainty interval of membership values are identified based on two different sources of imprecision: (i) selection of the level of fuzziness parameter, m, of IFC by identifying an m-bound (ii) determination of the list of optimum structures of fuzzy functions by identifying optimum forms of membership values.

IVT2FF is an iterative hybrid system, in which, the structure is learnt and parameters are tuned by a genetic learning algorithm, to determine the hidden structures viz. information points, which is the fundamental concept of the system identification. The ET2FF has three fundamental phases:

- **Phase 1**: Determination of the optimum uncertainty interval of the membership functions – FOU and optimum list of fuzzy functions and optimum values of other parameters with a soft computing algorithm. Here we use genetic learning process, although other optimization methods can be used as well.
- **Phase 2**: Type-2 FIS structure identification.
- **Phase 3**: Inference for testing dataset.

Phase 1: Genetic Learning Process (GLP). The idea is to create an optimization framework, using a soft computing method, e.g., Genetic Algorithms (GA) (Goldberg, 1989) to find the optimum system parameters and boundaries of the level fuzziness parameter to define boundaries for membership functions and the list of fuzzy functions that are most suitable for estimating local dependencies. Hence, the structure of each chromosome in GA framework encodes given type-2 FIS parameters, which are parameters of Improved Fuzzy Clustering (IFC) (Celikyilmaz & Turksen, 2008b) algorithm and fuzzy function structures. The parameter genes, in sequence, are composed of: two of the IFC clustering parameters, m-lower and m-upper $\in [1.01, \infty]$ and the type of the regression method, e.g. {$1='(linear\ regression)\ LSE'$, $2='(non\text{-}lienar\ regression)\ SVM'$, etc}, The rest of the parameter genes depend on the type of regression method. If SVM is used to construct more complex non-linear fuzzy functions, three additional SVM parameters, $Creg$, $epsilon$ and $kernel\ type$, are set up as additional alleles in the chromosome.

The rest of the nm different alleles represent the membership value transformations to be used to shape fuzzy functions. Among many different types, in our models we used power sets, exponential, sigmoid, logistic transformations, etc., of membership values as additional inputs. Each chromosome represents parameters of two separate models of type-1 FIS with Fuzzy Functions using two different m values, each of which has the same fuzzy function structure and regression parameters. Each individual in the population have different parameters and m boundaries so that population is diverse.

The optimum number of cluster, c^* is fixed based on $cviFF$ validity index of Fuzzy Function systems before GLP is processed. At the start of the GLP a wide range is assigned for the boundary values of m-interval, e.g.. {$m\text{-}lower=1.2$, $m\text{-}upper=7$}. For each chromosome, two separate type-1 FIS are constructed using each m-bound and parameters of the rest of the alleles.

In Fig. 3, FOU of the membership functions and fuzzy functions before and after GLP is shown. Note that these membership functions are the idealized representations of the membership values obtained from the IFC method. We do not curve fit the membership values into membership function in the actual calculations.
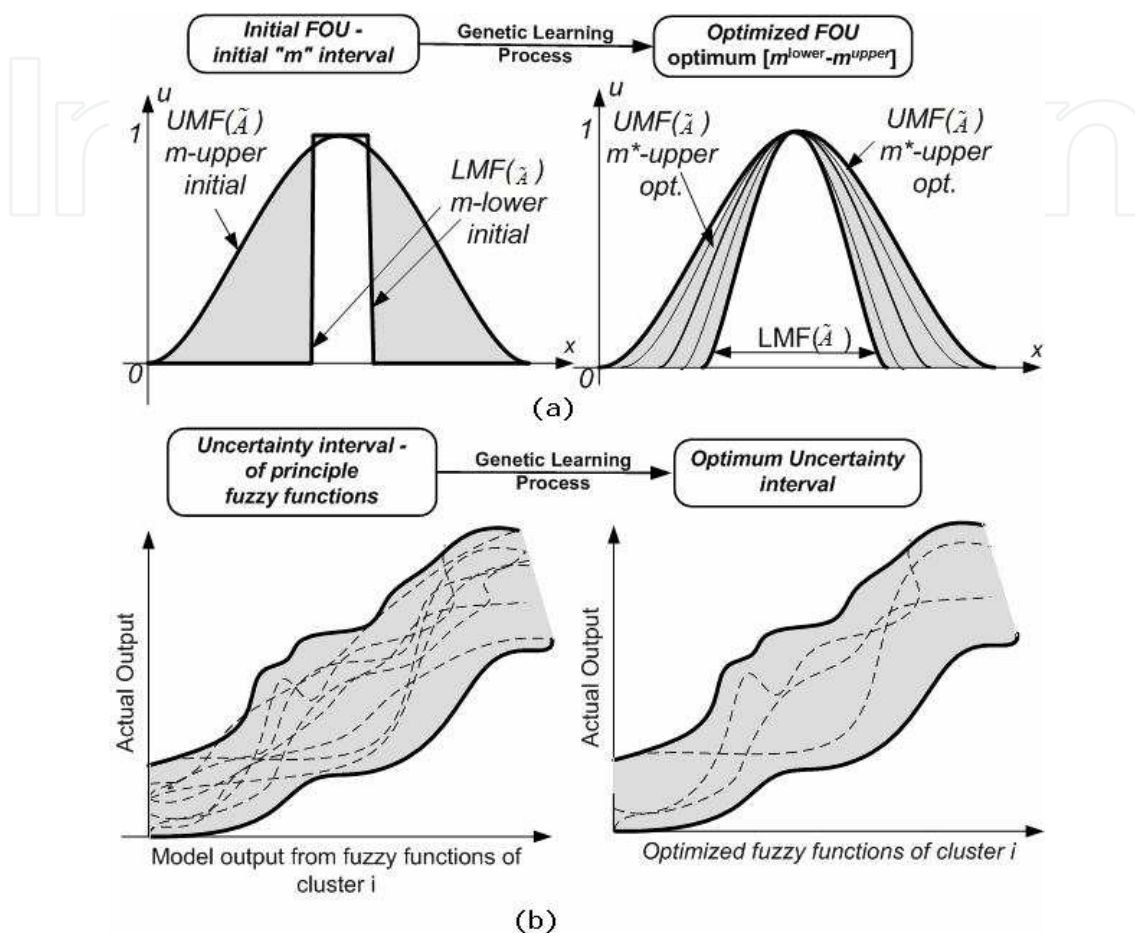

Fig. 3. Optimization using Genetic Learning Process. FOU of (a) idealized representation of the membership functions (MF), (b) output from principle fuzzy functions, UMF=Upper MF, LMF=Lower MF.

The membership functions, the top graphs, are predicted via IFC method. They are mainly based on two parameters, the level of fuzziness (m) and the structure of the interim fuzzy functions, $g_i(\tau_i)$, (as seen in (5) and (6)). The lower and upper membership functions-LMF($\tilde{A}$) and UMF($\tilde{A}$)- of the graph in Fig. 3.a on the left is formed using the initial *m-lower* and *m-upper* and the initial interim fuzzy function structures for the IFC method.

The interim fuzzy function parameters are randomly determined by the fuzzy function type and structure alleles (control genes) of each chromosome. They represent different forms of the membership values to be used to identify the interim fuzzy functions. In between the upper and lower boundaries of the shaded area- FOU any other type-1 membership value distribution can be formed using any value from [*m-lower, m-upper*] interval or any fuzzy function structure by combining different membership value transformations (Fig. 4). After IFC, two type-1 FIS are constructed using membership values and original input variables to build fuzzy functions to represent each local model.
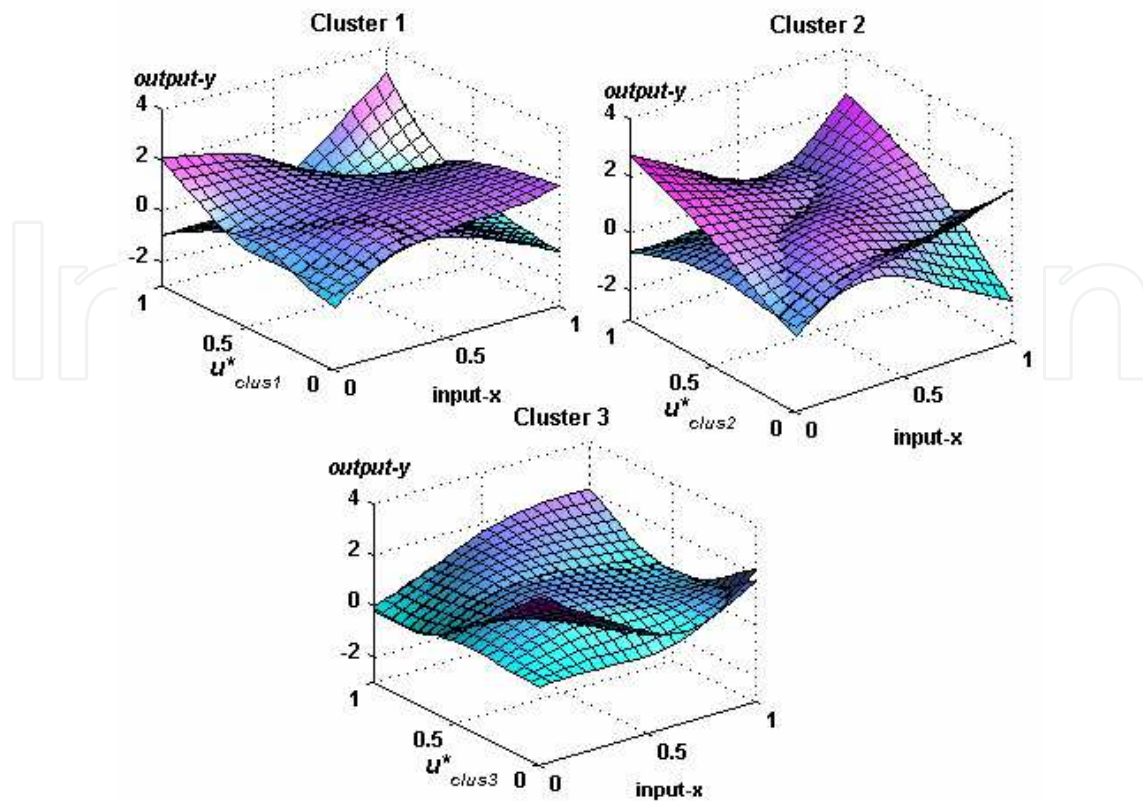
Fig. 4. Decision surfaces - $f(x,e^{\mu})$ obtained from GLP using parameters, SVM-Gaussian Kernel allele=0 {Non-linear} and $(m_{low}, m_{up}, Creg, \varepsilon)$={1.75,2.00,54.5,0.115}, c^*=3. u_{clusi} represents membership values of corresponding cluster i.

The algorithm starts with a larger interval of parameter values and optimizes the interval based on the fitness of each chromosome obtained from the combination of the boundary type-1 FISs. The fitness is evaluated as follows:

$$Fitness_p = -\sqrt{\frac{1}{n}\sum_{k=1}^{n}\left[(y_k - \hat{y}_{k,p}^{(m^{lower},\Omega)})^2 + (y_k - \hat{y}_{k,p}^{(m^{upper},\Omega)})^2\right]} \qquad (14)$$

'p' is the *population-size*, Ω is the optimum parameter list. The algorithm searches for the optimum model parameters and the *m-bound* so that the two type-1 FIS models would have the minimum error. Hence, the algorithm starts with a larger *m-bound* and gradually shifts to where the *Fitness_p* is maximized. To ensure that the fitness function increases monotonically, the best candidate solution in each generation enters the next generation directly.

Phase 2: Type-2 FIS Structure Identification. The optimum uncertainty intervals – FOU and the list of optimum fuzzy functions- determined in the previous step, are discretisized to find as many embedded type-1 FIS with fuzzy functions as feasible. The IVFF essentially is comprised of collection of embedded type-1 FISs.

Each embedded type-1 FIS defines a list of fuzzy functions for each cluster. These functions may or may not have the same input variables because each function of each cluster may be formed with a different membership value transformation used as additional inputs that best describes the local structure. Each fuzzy function would have a different membership

value as a variable and its different possible transformations to approximate the fuzzy functions. The algorithm presented here captures the best model parameters in cluster level among the embedded fuzzy models, one for each training vector, and keeps them in a matrix (collection table) to be used for reasoning.

Using the optimum parameters, from the previoys step the following steps are processed:

Step-1: The optimum m interval, [m-low^*,m-up^*] is discretisized into a list of disjoint m values. On the other hand, the optimum fuzzy function structures include information on different types of membership value transformations that can be used in formation of interim and principle fuzzy functions as additional inputs.

Step-2: For each combination of discrete parameters, IFC clustering is applied to partition the data into c^* clusters and calculate improved membership values. Membership values of the input space are calculated using IFC membership function in (6). For each discrete point x', different membership values are obtained from the IFC model using the list of learning parameter set.

Step-3: Fuzzy functions, $f_i^{r,s}$, $i=1,\ldots c^*$, of each embedded type-1 FIS model are determined using each set of discrete parameters and improved membership values using the functions such as in (8) depending on the model type.

For each cluster, only one of these approximated functions can explain the output better than rest of embedded functions. For instance, Fig. 5 depicts prediction performance of four different types of linear fuzzy functions of a single cluster using different m values based on root mean square error (RMSE). These four functions are formulized using different forms of membership value transformations shown in the label of in Fig.5. Every point corresponds to one function of a specific cluster. One specific model with a specific m value can reduce the error better than others. In another cluster, these results might be different and different fuzzy functions for different fuzziness levels could be more preferable. We need to determine the best functions obtained from different sets of parameters. This corresponds to finding the best embedded type-1 FIS model for each training vector using type-2 FIS system.
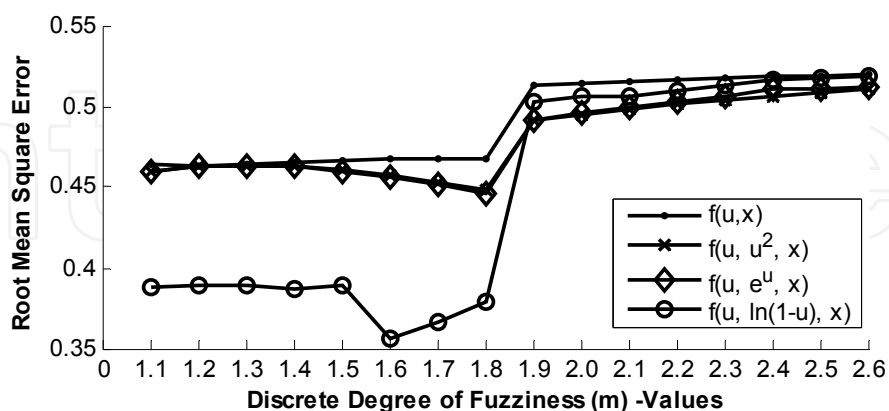

Fig. 5. The uncertainty in choosing the m values as a function of the error measure of the proposed type-2 FIS (ET2FF) - RMSE values as a function of degree of fuzziness (m) for four different fuzzy function structures. u: improved membership values.

Step-4: We find the parameters of each cluster that would give the minimum local fuzzy function error.

4.2.2 Full type-2 fuzzy functions

Interval type-2 fuzzy sets (IT2FS) are simplified forms of full type-2 fuzzy sets (FT2FS), where the secondary MEMBERSHIP FUNCTIONs are unified, e.g., equal to 1. Interval IT2FS identify footprint-of-uncertainty (FOU) as depicted in Fig. 6.
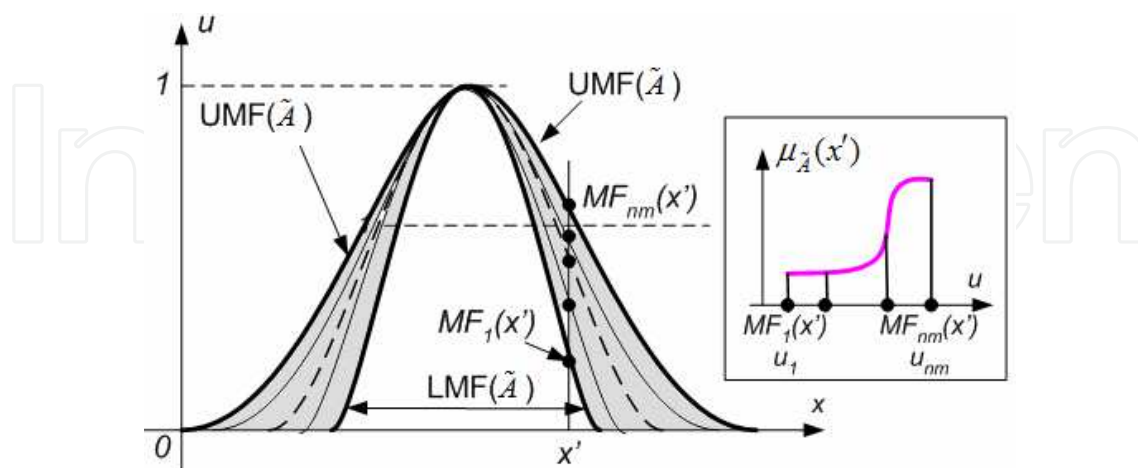

Fig. 6. Membership functions where base-end-points have uncertainty intervals. The insert represents secondary MEMBERSHIP FUNCTION of x'.

FOU of a FT2FS $\tilde{A}$ is the uncertainty region (2D-region) specified by lower and upper membership functions (membership functions), LMF($\tilde{A}$), UMF($\tilde{A}$). For each data point, x', there can be $nm=2,..,\infty$ different membership functions within this interval. Hence, FT2FS have secondary grades, which sit on top of FOU to form the 3D region.

In different studies, e.g., (Celikyilmaz & Turksen, 2008e;f), uncertainties of parameters from imperfect information are investigated using fuzzy clustering algorithm. In particular, the FOU of the IT2FS are formed based on the level of fuzziness parameter of FCM clustering.

In fuzzy clustering methods, fuzziness is measured by the level of fuzziness parameter, m, which determines the degree of overlap between the clusters, viz. structures, granules, etc., identified in the given dataset. In many research, identification of the footprint_of_uncertainty of membership functions of FCM clustering algorithm, e.g., (Hwang & Rhee, 2007; Celikyilmaz & Turksen, 2008e), or hybrid clustering algorithms (Celikyilmaz & Turksen, 2008f) is based on the level of fuzziness parameter. One can investigate the level of fuzziness, m, of particularly fuzzy c-regression model (FCRM) clustering methods (Hathaway & Bezdek, 1993), instead of conventional clustering algorithms. In building fuzzy inference systems, separate functions are identified for each local input-output relation, which are defined with hyperplanes. Therefore, a better way is to construct hyperplane-shaped clusters.

Thus, we presented a new type-2 fuzzy inference method (Celikyilmaz & Turksen, 2008g), which can identify the optimum secondary membershp function grades, i.e., weights, of the primary MF grades using genetic algorithms. New data vectors adopt the secondary membership function grades obtained from the training samples in their neighborhood. During genetic learning process, each individual in the population encodes these weights for each training vector for each cluster, separately. This is quite cumbersome process when the number of training vectors are large therefore it is simplified in this paper by implementing transductive learning method. Instead of learning the secondary MF grades of the entire training dataset, for each new data point a new set of weights are learnt from

fairly less training vectors, which are close to this new vector in distance. Experimental analysis demonstrates the performance of the new approach.

The distibution of secondary membership functions is demonstrated in Fig. 7 using an artificial dataset. The dataset ontains single input and single output with two local structures; therefore, the number of clusters is set to two. The primary MF grades, $u(x)$ values, are obtained from FCRM model using list of levels of fuzziness parameter $m=\{1.1,1.25,..,2.6\}$ as shown in Fig. 7 top-right graph, also the base of the 3D graph , the bottom graph in Fig. 7. The bottom 3-D graph in Fig. 7 displays secondary membership function of a single point $x_k=0.5$. The secondary membership function values of nearest data points are optimized with genetic algorithms.
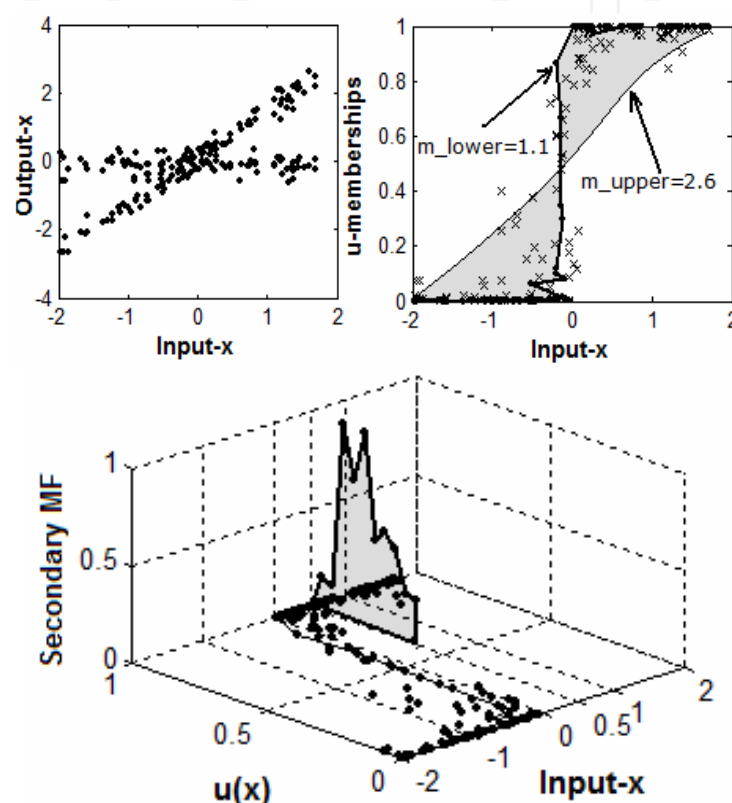
Fig. 7. (Top-left) Artificial Dataset, (Top-right) FOU by $m \in [1.1, 2.6]$, (Bottom) secondary MF of data point $x = 0.5$.

5. Experiments on text mining

In this paper we present various different fuzzy function approaches which is a summary of our research for the last five years. Our experiments have shown that as we introduce the uncertainty, we gain more performance from the models that we build to represent the real systems, i.e., variaous natual language processing applications on infomration retrieval and information extraction. Hence, the interval type-2 fuzzy system models based on fuzzy functions have shown better performance improvement compared to the type-2 fuzzy function models (Celikyilmaz & Turksen, 2008a). Later on we have developed the full type-2 fuzzy functions method with which we can introduce second-order uncertainties to the system model. The results have shown that the full type-2 fuzzy functions can improve the

performance of fuzzy system models when there is uncertainty. Since natural language appliations are imprecise in nature, we prefer to use full type-2 fuzzy functions when building language models. In addition the space limitations keep us presenting all the result from different our different system modeling approaches. Hence, in the next we will present the result of our experiments using Full Type-2 Fuzzy Functions, in other words, Type-2 Fuzzy Inference System (T2FIS) presented in this paper. We will build a Question and Answering (QA) system.

The aim of QA systems is to find precise answers to natural language questions from large document collections by processing several modules in sequence including question analysis, document retrieval, answer extraction and answer selection. In this paper we are particularly interested in answer selection part, in which retrieved candidate answers are ranked based on a *textual entailment* model[1]. An entailment relation between two text snippets (text-hypothesis pair) is produced when the meaning of the hypothesis meaning can be inferred from the meaning of text.

Inasmuch our QA system is designed to return candidate sentences from a corpus, instead of returning exact answer phrases, such as we return the sentence containing the answer-phrase but not extract the phrase[2]. Hence, we try to find binary entailment relationships between queries and candidate sentences with the hypothesis that the answer phrase is likely to be contained in them. Firstly, we convert a question into a regular sentence, which represents our hypothesis sentence to be entailed (hypothesis-*h*) and then use *textual entailment* module to identify if the candidate sentence (text-*t*) entails *h*. Thus, given a (*t-h*) pair, we try to recognize the relation between the meaning of the text and hypothesis as a *true entailment* if the meaning of the hypothesis is entailed from the meaning of the text such as given follows:

<p style="text-align:center;">t: Harry was born in Iowa.
h: Harry's birthplace is Iowa.</p>

t entails *h,* otherwise we recognize the relation between the meaning of the texts as *false entailment*. In this section, we demonstrate experiments conducted on Textual Entailment datasets (freely available from PASCAL recognizing textual entailment (RTE) challenge conference- http://pascallin.ecs.soton.ac.uk/Challenges/RTE/) using the proposed T2FIS method. The goal of RTE challenge is to recognize semantic inference that a textual entailment defines directional relation between two text fragments, called *text* (*T*) and *hypothesis* (*H*) so that a human being can infer that *H* is most likely true on the basis of the contents of *T*. As a further note, we use the entailment model to build a QA system.

Using the RTE datasets, we build a classifier model using proposed T2FIS method. This model is build to be implemented to our Question Answering (QA) system to rank the sentences retrieved from a search engine while matching each retrieved sentence (T) with the question query sentence (H). The question query is transformed into a sentence putting a placeholder to where the answer should be in the question sentence.

---

[1] Textual entailment models are first introduced in Pascal RTE conference (Dagan et.al. 2006).

[2] Answer-extraction is left out as a future research study on natural language processing applications.

Dataset. There are four different RTE challenges so far, each having different sets of T-H pairs. We combined the first three RTE datasets and only used the *T-H* pairs that are specifically designed for QA systems, i.e., there are different sets of pairs constructed for different applications such as summarization, information retrieval, etc. The hypothesis in T-H pairs are formed by converting a question sentence into a regular sentence and placing the question word (what, how, when, etc.) with the correct/false answer as shown in Table 4.

Example Pairs	*Entailment*
T: In February 2002, President George Bush visited China to mark the 30th anniversary of Nixon's historic trip. **H**: Nixon visited China in February 2002.	*FALSE*
T: The Chernobyl nuclear-power plant is in Ukraine, but the reactor that exploded during the night of April 26, 1986, is only 10 miles from the Belarusian border **H**: The Chernobyl disaster took place on the 26th of April, 1986.	*TRUE*
T: Microsoft was established in Italy in 1985. **H**: Microsoft was established in 1985.	*FALSE*

Table 4. Examples of Text-Hypothesis pairs from Recognizing Text Entailment Challenge.

Features: We extract different sets of attributes from the *T-H* pairs and to generate some of these features, we used different tools including Stanford Tagger (Klein and Manning, 2003), Named Entity Tagger (Finkel et.al., 2005), WordNet::Similarity Package (WordNet). Each (T-H) pair is analyzed to extract features (input variables), which depend on the relation between them, some of which is shown as follows:

Lexico-Syntactic Overlap-Alignment Features: These features range from the ratio of the consecutive word overlap between the T and H (*n-gram, i.e., $n \in \{1,2,3\}$*), the lowest common subsequence which measures the similarity between text T with length *m* and hypothesis H with length *n*, by searching in-sequence matches that reflect sentence level word order. Other features in this category are skip-*n*gram, number of common pair of words in T and H in order with gaps.

Semantic Features: Noun, verb and adjective/adverb specific semantic overlap metric (similarity measure) using WordNet's hypernym, hyponym, negation match between T-H based on clue phrases such as 'no', 'not', 'neither', etc., which are some of the examples of the features extracted from T-H pairs.

Since the task is text entailment, we extracted two verb match statistics using WordNet's *cause to* and *entailment* relations. For each verb pair that groups a verb from the text v_T and one from the hypothesis v_H we tested either a caused by or entailment relation when;

- *verb entailment:* v_H entailment v_T
- *verb cause:* v_T cause to v_H

To generate separate features for each relation, we counted the number of verb pairs constructed in the above form.

We generate the train and testing datasets using the T-H pairs from RTE challenge and extract features as explained above to form the inputs. The binary output variable having

the value '1' indicates "true entailment" and '0' "false entailment", and these are assigned manually (given by the RTE challenge datasets). We extract 29 features using different combinations of the lexico-syntactic and semantic features. We use 1670 T-H pairs for building the learning models--training and 2400 pairs for testing purposes. False and true entailments are evenly distributed. We used 10, 50, 100, 200 number of training vectors as the number of nearest neighbors to build four different T2FIS models, i.e., T2FIS_10, T2FIS_50, T2FIS_100, T2FIS_200, and analyzed the difference in the experiments. For the rest of the benchmark models, we randomly selected 750 training samples five times to build different models and analyzed their average testing performance on the same testing dataset and measured their error margins between five experiments, i.e., standard deviation of the accuracies.

Model Construction. The system model performance is measured with accuracy,

$$accuracy(\%) = \frac{(True\ Positives) + (True\ Negatives)}{number\ of\ data(nd)}$$

Since the classification model outputs are probabilities, different threshold values (to discern between two classes) values are varied to obtain the optimum True Positives (TPs) and True Negatives (TNs) during learning stage of each modeling approach. The threshold values that are identified by the structure identification are used during inference to estimate class labels of testing dataset. The same parameters that are used in the previous experiments is used in this experiments as well with the exception that the algorithms are designed to find classifier functions, e.g., SVM for classification, and FCCM of T2FIS methods are used. The feature extraction, explained above, is implemented as a part of entailment into our QA system using Java and the T2FIS is implemented using Matlab. The average accuracy results from the five repetitions of experiments and the model with the best average accuracy are shown in Table 5.
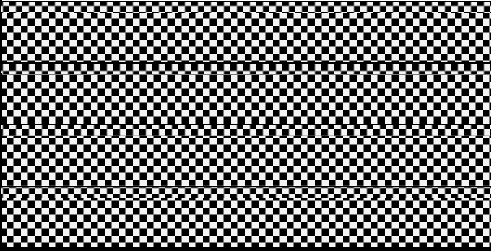
Model -- accuracy	*Testing Dataset Average Accuracy*	*significance-test between the best T2FIS model and the benchmarks (p<0.05)*
T2FIS-10	0.579	
T2FIS-50	0.585	
T2FIS-100	**0.598**	
T2FIS-200	0.582	
ANFIS	0.547	0.001
SVM-LIN	0.568	0.021
SVM-RBF	0.561	0.009
NN	0.550	0.006

Table 5. Accuracy results of the Text Entailment for QA tasks.

Based on the results of this experiment, the best testing accuracy is obtained when the T2FIS is executed for when 100 nearest training vectors are used (T2FIS_100). Compared to the benchmark methods, there is a [5-9%] improvement when the proposed T2FIS_100 is used. Fig. 8 shows accuracies along with their standard deviations across five separate experiments.
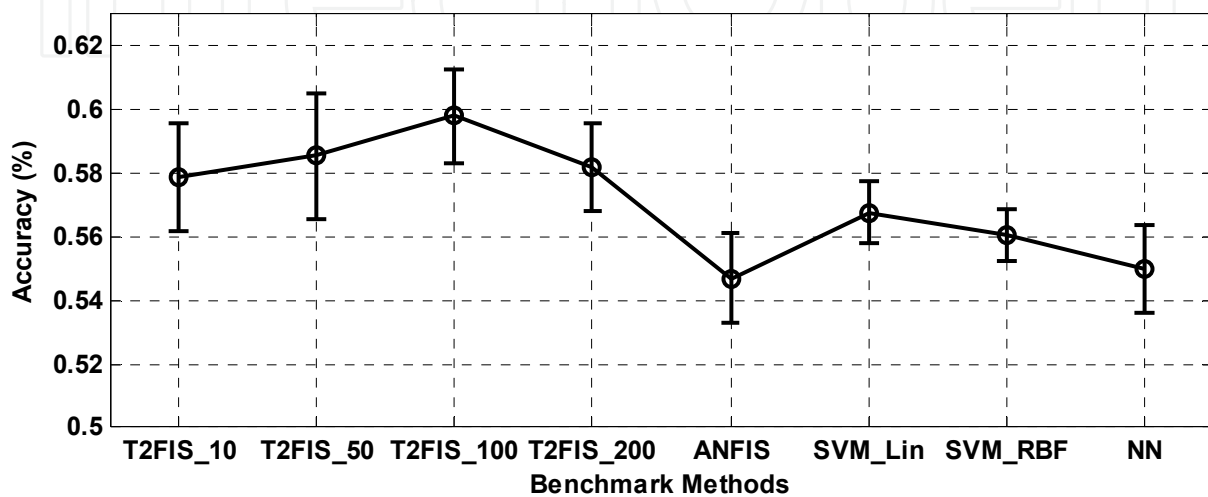

Fig. 8. Accuracy Comparison Graph between the models constructed based on five repetitions. The standard errors are also shown as error-bars.

We also measure the statistical significance of the proposed approach T2FIS on classification problems. The same two-sample left-tailed *t-test* with 95 percent confidence level is used to indicate the significance of the optimum models of each methodology. Our hypothesis is that cross validation errors of the best proposed method (T2FIS_100) and the rest of the models are same with 95% confidence. In Table 5 *significance probabilities* are shown. In all experiments the T2FIS_100 model is significantly better than the benchmark models ($p<0.05$). Thus, we can conclude that the proposed algorithm has comparable/significantly better results than other powerful well-known modeling tools.

The proposed approach helps us to quantify the uncertainty in the membership functions used in the fuzzy system models and variation in model parameters. It is shown that it is possible to capture the variations with a list of discrete membership functions and weighing them individually to incorporate their individual effects to the model. We quantify this uncertainty based on the imprecision in the level of fuzziness parameter of the fuzzy clusters we identify. The real problems can be modeled by using type-2 membership functions which can be derived from the changing values of the fuzziness of the clusters. Hence, when the expert is not present to identify the fuzzy sets, this method could provide better solutions. With the two experiments, we showed that the T2FIS is better compared to the rest of the fuzzy or non-fuzzy reasoning approaches based on the modeling error.

6. Conclusions

Fuzzy logic encompasses conceptual framework of sets and logic that is able to handle both precise and imprecise information and meaning. Although fuzzy systems still do not necessarily outperform human in dealing with uncertainty and imprecision, it helps to reduce the real world problems to a scale that is possible for computing solutions that was impossible before. The principal objective of the presented methods of this paper is to develop applications to enable information extraction under uncertainty, particularly on the conception and design of autonomous systems for natural language processing applications specifically on question and answering systems and textual entailment mechanism. A direct practical application of fuzzy logic to these fields does not seem to exist at present. Thus, higher order fuzzy system models based on fuzzy functions will have many uses in textual and semantic analysis, data mining, and search algorithms in the near future.

In this paper, a new approach to information extraction via fuzzy functions is presented. The presented type-2 fuzzy inference system is used for uncertainty quantification of real-world data. Partitioning a given set of data into granules is most fundamental problem in pattern recognition and data mining. With the presented fuzzy inference system, we define membership functions based on the given dataset and use fuzzy clustering methods. The approach of membership function elicitation of type-2 fuzzy inference system is a category of hybrid knowledge-data class of fuzzy set elicitation and enables employment of different membership functions and local dependency function structures for each cluster. The major benefit of this approach is that, it does not require definition of membership function by an expert. The primary membership functions are found from fuzzy clustering methods presented in the paper and the secondary membership grades are optimized with genetic algorithms. The algorithm adopts simple type-reduction and does not require defuzzification. Textual entailment task is a challenging problem and depends on careful analysis of the features between the question and candidate answer pairs and an efficient classifier model such as uncertainty modeling tool presented in this paper.

7. References

Bandler, R. and Grinder, J. (1996). *The Structure of Magic: A Book about Language and Therapy.* 198 pages, Science and Behaviour Books, ISSN 0831400447.

Babuska, R. and Verbruggen, H.B. (1997). Constructing fuzzy models by product space clustering. In *H. Hellendoorn and D Driankow(Eds.) Fuzzy Model identification: Selected Approaches*, 53-90, Springer, Berlin, Germany, 1997.

Bezdek, J.C., Ehrlich, R. and Full, W. (1984) FCM: Fuzzy C-Means Algorithm. *Computers and Geoscience*, 10, (1984) 191-203.

Bodur, M. and Acan, A., Unveren, A. (2006). Reduction of Generalization error in Fuzzy System Models. *2006 IEEE Int. Conf. on Fuzzy Systems, Vancouver, BC.*, 2184-2189. ISBN 0-7803-9488-7.

Celikyilmaz, A. and Turksen, I.B. (2009a). Increasing Accuracy of Two Class Pattern Recognition with Improved Fuzzy Functions. Expert Systems with Applications, 36, (March 2009) 1337-1354, ISSN 0957-4174.

Celikyilmaz, A. and Turksen, I.B. (2009b). Genetic Fuzzy System Based on Improved Fuzzy Functions. Journal of Computers, 4, 2, (February 2009) 135-146.

Celikyilmaz, A. and Turksen, I.B. (2009c). Spectral Learning with Type-2 Fuzzy Numbers for Question/Answering System. *IFSA World Congress, EUSFLAT Conference, Lisbon, Portugal*, July 2009.

Celikyilmaz, A. and Turksen, I.B. (2009d). Kernel Based Hybrid Fuzzy Clustering for Non-Linear Fuzzy Classifiers, *Proc. 27th Intern. Conf. of the North American Fuzzy Information Processing Society – NAFIPS- IEEE*, (June 2009).

Celikyilmaz, A. and Turksen, I.B. (2008a). Uncertainty modelling with evolutionary improved fuzzy functions approach. *IEEE Systems, Man, and Cybernetics- Part B*, 38, 4, (August 2008) 1098-1110, ISSN 1083-4419.

Celikyilmaz, A. and Turksen, I.B. (2008b). Enhanced Fuzzy System Models with Improved Fuzzy Clustering Algorithm. *IEEE Trans. on Fuzzy Systems*, 16, 3, (June 2008) 779-794, ISSN 1063-6706.

Celikyilmaz, A. and Turksen, I.B. (2008c). Validation Criteria for Enhanced Fuzzy Clustering. *Pattern Recognition Letters*, 29, 2, (January 2008) 97-108, ISSN 0167-8655.

Celikyilmaz, A. and Thint, M. (2008). Semantic Approach to Textual Entailment for Question Answering – A New Domain for Uncertainty Modeling. *7th IEEE Int. Conf. Cognitive Informatics (ICCI'08), Stanford University, CA, IEEE CS Press*, (August 2008) 481-487, ISBN 978-1-4244-2538-9.

Celikyilmaz, A. and Turksen, I.B. (2008d). Type-2 Fuzzy Classifier Ensembles for Text Entailment. *Joint Conference on Information Sciences (JCIS2008)– Fuzzy Theory and Technology Track, Shenzhen, China*, (December 2008) 1-7, ISSN 1951-6851.

Celikyilmaz, A. and Turksen, I.B. (2008e). Uncertainty Bounds of Fuzzy C-Regression Method. *IEEE Int. Conference on Fuzzy Systems - World Congress on Computational Intelligence (WCCI 2008), Hong Kong*, (June 2008) 1193-1198, ISSN 1098-7584.

Celikyilmaz, A. and Turksen, I.B. (2008f). Genetic Type-2 Fuzzy Classifier Functions. *Proc. 26th Intern. Conf. of the North American Fuzzy Information Processing Society – NAFIPS- IEEE*, (May 2008) 1-6, ISBN 978-1-4244-2351-4.

Celikyilmaz, A. and Turksen, I.B. (2008g). A Type-2 Fuzzy C-regression Method. *Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU-2008), Malaga, Spain*, (2008).

Celikyilmaz, A. and Turksen, I.B. (2007). Fuzzy Functions with Support Vector Machines. *Information Sciences*, 177, 23, (December 2007) 5163-5177, 2007.

Chen, J.-Q. and Xi, Y.-G., and Zhang, Z.-J. (1998) A clustering algorithm for fuzzy model identification. *Fuzzy Sets and Systems*, 98, 3, (Sept. 1998) 319-329, ISSN 0165-0114.

Coupland, C. and John, R. (2007) Geometric Type 1 and Type 2 Fuzzy Logic Systems. IEEE Trns. Fuzzy Systems, 15, (February 2007) 3-15.

Dogan, I. and Glickman, O. And Magnini, B. (2006) The PASCAL Recognising Textual Entailment Challenge. *In Quinonero-Candela, J. and Dogan, I., and Magnini, B.*

And D'Alche-Buc, F. (Eds), Machine Learning Challenges. LNCS 3944 (2006) 177-190.

Demirci, M. (1999). Fuzzy functions and their fundamental properties. *Fuzzy Sets and Systems*, 106, 2, (September 1999) 239-246, ISSN 0165-0114.

Emami, M.R. and Turksen, I.B., Goldenberg, A.A. (1998). Development of a Systematic Methodology of Fuzzy Logic Modelling. *IEEE Trans. on Fuzzy Systems*, 63, (August 1998) 346-361, ISSN 1063-6706.

Finkel, J.R. and Grenager, T. And Manning, C. (2005) Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. Proc. 43nd Annual Meeting of the Assiciation for Computational Linguistics (ACL 2005) (2005) 363-370.

Goldberg, D.E. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison Wesley Publishing.

Gunn, S. (1998). Support Vector Machines for Classification and Regression. *ISIS Technical Report*.

Hagras, H. (2004). A Hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots. *IEEE Trans. Fuzzy Syst.*, 14, 4, (Aug. 2004) 524-539, ISSN 1063-6706.

Hathaway, R. and Bezdek, J. (1993) Switching regression model and fuzzy clustering. *IEEE Trans. Fuzzy Syst.* 1, 195-204.

Höppner, H. and Klawonn, F. (2003) Improved fuzzy partitions for fuzzy regression models. *Int. Jrnl. of Approximate Reasoning*, 32, (2003) 85-102, ISSN 0888-613X.

Hwang, C. and Rhee, F.C.-H. (2007) Uncertain fuzzy clustering: interval type-2 fuzzy approach to c-means. *IEEE Trans. Fuzzy Syst.* 15, 107-120.

Klein, D., and Manning, C.D. (2003) Fast Exact Inference with a Factored Model for Natural Language Parsing. *In Advances in Neural Information Processing Systems (NIPS 2002)*, 15 (2003) 3-10.

Menard, M. (2001). Fuzzy Clustering and Switching Regression Models Using Ambiguity and Distance Rejects. *Fuzzy Sets and Systems*, 122, (2001) 363-399.

Mendel, J.M. (2001). *Uncertain Rule-Based Fuzzy Logic Systems: Theory and Design.* Prentice, Upper Saddle River, 2001.

Mizumoto, M. (1989). Method of Fuzzy Inference suitable for Fuzzy Control. *J. Soc. Instrument Control Engineering*, 584, (1989) 959-963.

Turksen, I.B. (2008). Fuzzy Functions with LSE. *Applied Soft Computing*, 8,3, (June 2008) 1178-1188, ISSN 1568-4946.

Turksen, I.B. and Celikyilmaz, A. (2006). Comparison of Fuzzy functions with Fuzzy Rule Bases. *Int. Journal of Fuzzy Systems*, 8, 3, (2006) 137-148.

Sasaki, M. (1993) Fuzzy Functions. *Fuzzy Sets and Systems*, 55, 3, (May 1993) 295-301, ISSN 0165-0114.

Takagi, T. and Sugeno, M. (1985). Fuzzy Identification of Systems and Its Applications to Modeling and Control. *IEEE Transactions on Systems, Man and Cybernetics,* 15, 1, (1985) 116-132, 1985.

Uncu, O. and Kilic, K. and Turksen, I.B. (2004). A new fuzzy inference approach based on Mamdani inference using discrete type-2 fuzzy sets. *IEEE Intrn. Conf. Syst.,Man, Cbyern., B, Cybern.,* 3, (October 2004) 2272-2277.

WordNet, http://wordnet.princeton.edu/perl/webwn?s=word-you-want

Zadeh, L.A. (1965) Fuzzy Sets. *Inform.& Control*, 8 (1965) 338-353.

Fuzzy Systems

Edited by Ahmad Taher Azar

While several books are available today that address the mathematical and philosophical foundations of fuzzy logic, none, unfortunately, provides the practicing knowledge engineer, system analyst, and project manager with specific, practical information about fuzzy system modeling. Those few books that include applications and case studies concentrate almost exclusively on engineering problems: pendulum balancing, truck backeruppers, cement kilns, antilock braking systems, image pattern recognition, and digital signal processing. Yet the application of fuzzy logic to engineering problems represents only a fraction of its real potential. As a method of encoding and using human knowledge in a form that is very close to the way experts think about difficult, complex problems, fuzzy systems provide the facilities necessary to break through the computational bottlenecks associated with traditional decision support and expert systems. Additionally, fuzzy systems provide a rich and robust method of building systems that include multiple conflicting, cooperating, and collaborating experts (a capability that generally eludes not only symbolic expert system users but analysts who have turned to such related technologies as neural networks and genetic algorithms). Yet the application of fuzzy logic in the areas of decision support, medical systems, database analysis and mining has been largely ignored by both the commercial vendors of decision support products and the knowledge engineers who use them.

INTECH
open science | open minds