Available online at www.sciencedirect.com
ScienceDirect
Information Sciences 178 (2008) 381–402
INFORMATION SCIENCES
AN INTERNATIONAL JOURNAL
www.elsevier.com/locate/ins

A vector similarity measure for linguistic approximation: Interval type-2 and type-1 fuzzy sets

Dongrui Wu, Jerry M. Mendel *

Signal and Image Processing Institute, Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089-2564, USA

Received 7 October 2006; received in revised form 3 March 2007; accepted 30 April 2007

Abstract

Fuzzy logic is frequently used in *computing with words* (CWW). When input words to a CWW engine are modeled by interval type-2 fuzzy sets (IT2 FSs), the CWW engine's output can also be an IT2 FS, $\widetilde{A}$, which needs to be mapped to a linguistic label so that it can be understood. Because each linguistic label is represented by an IT2 FS $\widetilde{B}_i$, there is a need to compare the similarity of $\widetilde{A}$ and $\widetilde{B}_i$ to find the $\widetilde{B}_i$ most similar to $\widetilde{A}$. In this paper, a vector similarity measure (VSM) is proposed for IT2 FSs, whose two elements measure the similarity in shape and proximity, respectively. A comparative study shows that the VSM gives more reasonable results than all other existing similarity measures for IT2 FSs for the linguistic approximation problem. Additionally, the VSM can also be used for type-1 FSs, which are special cases of IT2 FSs when all uncertainty disappears.

Keywords: Similarity measure; Compatibility measure; Type-1 fuzzy set; Interval type-2 fuzzy set; Computing with words; Linguistic approximation

1. Introduction

Zadeh coined the phrase "*computing with words*" (CWW) [48,49]. According to him, CWW is "*a methodology in which the objects of computation are words and propositions drawn from a natural language*". It is "*inspired by the remarkable human capability to perform a wide variety of physical and mental tasks without any measurements and any computations.*" Nikravesh [35] further pointed out that CWW is "*fundamentally different from the traditional expert systems which are simply tools to 'realize' an intelligent system, but are not able to process natural language which is imprecise, uncertain and partially true.*"

Our thesis is that *words mean different things to different people* and so there is uncertainty associated with words, which means that fuzzy logic must somehow use this uncertainty when it computes with words [25,26].

* Corresponding author. Tel.: +1 213 740 4445; fax: +1 213 740 4651.
E-mail addresses: dongruiw@usc.edu (D. Wu), mendel@sipi.usc.edu (J.M. Mendel).

0020-0255/$ - see front matter
doi:10.1016/j.ins.2007.04.014

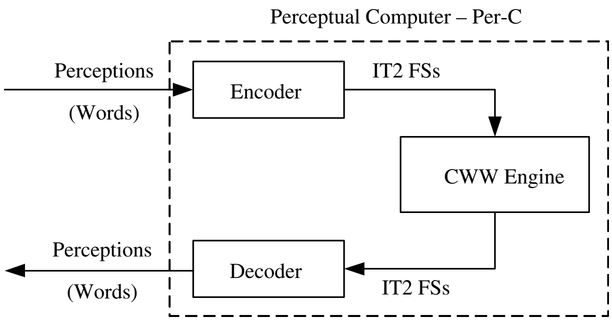

Fig. 1. Conceptual structure of CWW.

Hence, we argue that interval type-2 fuzzy sets (IT2 FSs) should be used in CWW [28]. We will limit our discussions to IT2 FSs in this paper.

A specific architecture is proposed in [27] for making judgements by CWW. A slightly modified architecture is shown in Fig. 1. It will be called a *perceptual computer* – Per-C for short. Perceptions (i.e., granulated terms, words) activate the Per-C and are also output by the Per-C; so, it is possible for a human to interact with the Per-C just using a vocabulary of words. In Fig. 1, the *encoder*[1] transforms linguistic perceptions into IT2 FSs that activate a *CWW engine*. The *decoder*[2] maps the output of the CWW engine into a word. Usually a vocabulary (codebook) is available, in which every word is modeled as an IT2 FS. The output of the CWW engine is mapped into a word (in that vocabulary) most similar to it.

The CWW engine, e.g. rules, the linguistic weighted average (LWA) [43], etc., maps IT2 FSs into IT2 FSs. If the CWW engine is rule-based, its output may be a crisp number (e.g., after defuzzification), in which case the decoder can map this number into a word in the vocabulary, as explained in [27]. On the other hand, if the CWW engine uses the LWA, its output is an IT2 FS $\tilde{A}$, or if the CWW engine is rule-based, but its output is also an IT2 FS $\tilde{A}$, then the decoder must also map $\tilde{A}$ into a word in the vocabulary. In this paper it is assumed that the output of the CWW engine is an IT2 FS $\tilde{A}$.

How to transform linguistic perceptions into IT2 FSs, i.e. the encoding problem, has been considered in [30–32]. This paper considers the decoding problem, or, as called by Zadeh [48,49], *linguistic approximation*, i.e. how to map an IT2 FS $\tilde{A}$ into a word (linguistic label). More specifically, given a vocabulary consisting of N words with their associated IT2 FSs $\tilde{B}_i$ $(i = 1, \ldots, N)$, our goal is to find the $\tilde{B}_i$ which most closely resembles $\tilde{A}$, the output of the CWW engine. The word associated with that $\tilde{B}_i$ will then be viewed as the output of the Per-C. To do this, it must be possible to compare the similarity between two IT2 FSs. A vector similarity measure (VSM) for IT2 FSs is proposed in this paper.

The rest of this paper is organized as follows: Section 2 gives the definitions of similarity, proximity and compatibility, which are closely related to each other. Section 3 reviews four existing similarity measures for IT2 FSs. Section 4 proposes a VSM for IT2 FSs. Section 5 provides discussions on a number of issues and shows that the VSM for IT2 FSs can also be used for type-1 (T1) FSs when all uncertainty disappears. Section 6 draws conclusions. Some background material about IT2 FSs is given in Appendix A. Proofs of the theorems are given in Appendix B.

2. Definitions

Similarity, proximity and compatibility are three closely related concepts. There are different definitions on the meanings of them [8,12,20,24,38,45,46]. According to Yager [45], a proximity relationship between two T1 FSs A and B on a domain X is a mapping p: $X \times X \to T$ having the properties: (1) *Reflexivity:* $p(A, A) = 1$; and, (2) *Symmetry:* $p(A, B) = p(B, A)$. Often T is the unit interval.

[1] Zadeh calls this *constraint explicitation* in [48,49]. In [50] and some of his recent talks, he calls this *precisiation*.

[2] Zadeh calls this *linguistic approximation* in [48,49].