IOWA & Cross-ratio Uninorm operators as aggregation tools in sentiment analysis and ensemble methods

Orestes Appel
Centre for Computational Intelligence
De Montfort University, UK[1]
Bissett School of Business
Mount Royal University, Canada[2]
Email[1]: orestes.appel@my365.dmu.ac.uk
Email[2]: oappel@mtroyal.ca

Francisco Chiclana and Jenny Carter
Centre for Computational Intelligence
De Montfort University, Leicester, UK
Email: {chiclana, jennyc}@dmu.ac.uk

Hamido Fujita
Iwate Prefectural University
Takizawa, Iwate, Japan
Email: issam@iwate-pu.ac.jp

Abstract—**In the field of Sentiment Analysis, a number of different classifiers are utilised to attempt to establish the polarity of a given sentence. As such, there could be a need for aggregating the outputs of the algorithms involved in the classification effort. If the output of every classification algorithm resembles the opinion of an expert in the subject at hand, we are then in the presence of a group decision-making problem, which in turn translates into two sub-problems: (a) defining the desired semantic of the aggregation of all opinions, and (b) applying the proper aggregation technique that can achieve the desired semantic chosen in (a). The objective of this article is twofold. Firstly, we present two specific aggregation semantics, namely fuzzy-majority and compensatory, which are based on Induced Ordered Weighted Averaging and Uninorm operators, respectively. Secondly, we show the power of these two techniques by applying them to an existing hybrid method for classification of sentiments at the sentence level. In this case, the proposed aggregation solutions act as a complement in order to improve the performance of the aforementioned hybrid method. In more general terms, the proposed solutions could be used in the creation of semantic-sensitive ensemble methods, instead of the more simple ensemble choices available today in commercial machine learning software offerings.**

I. Introduction

The idea of combining in a sensible manner the output of a number of machine learning algorithms, instead of relying on the results provided by the participating methods in isolation, was brought to the research community a few years ago. Rokach [10] provide the following motivation for doing so: “The main idea behind the ensemble methodology is to weigh several individual classifiers, and combine them in order to obtain a classifier that outperforms every one of them. In fact, human being tends to seek several opinions before making any important decision. We weigh the individual opinions, and combine them to reach our final decision” [10]. In an ensemble method for classification purposes the following building blocks can be identified [10]: (i) a training set, (ii) a base inducer (forms a classifier that represents the generalized relationship between the input attributes and the target attribute), (iii) a diversity generator (of diverse classifiers), and (iv) a *combiner* (of the classifications outputs of the classifiers).

This article focuses on aggregation techniques that could be used as the *combiner* component of the outputs of a number of classification method trained by the same dataset, or it could simply be the outputs of a mix of supervised machine learning and lexicon-based methods in the context of Sentiment Analysis (SA). The two algorithms proposes are based on the Induced Ordered Weighted Averaging (IOWA) operator and the Cross-ratio Uninorm operator.

The rest of this article is organised as follows: Section II presents some related work; Section III describes the IOWA operator approach; Section IV addresses the cross-ratio compensative uninorm approach and its implementation in the SA problem; Section V presents some experimental results that prove that the proposed aggregation techniques actually produce expected outputs; Section VI provides more details on utilising the proposed approaches in ensemble methods; and finally, Section VII shares some conclusions and suggests some possible areas of future research.

II. Related work

Typically, the importance of the values being fused is application-dependent and therefore “appropriately combining the individual argument weights with the characterizing weights of the operator to obtain operational weights to be used in the actual aggregation” [19] is key. In the field os SA, which is relevant to this article, Perkins [9] propose to use median, voting and arithmetic mean when aggregating multiple classification results. In the field of soft-computing, the study of effective mechanisms for aggregation has also been a central part of research. In this field, fuzzy sets methods play a key role in areas of particular interest for us such as data fusion and group decision-making. In the latter, the clear intention is to combine in a meaningful way the opinion of a number of individuals or methods. In [11] Rudas, Pap & Fodor show how key information fusion is in many complex areas like decision making, utility theory, fuzzy inference systems, robotics and vision.

The work of Peláez et al. [8], on OWA operators in decision-making aimed to obtaining the opinion of the majority, is very influential. In [12], Wu and Chiclana derived the mathematical modelling and representation of the multiplicative transitivity property to the case of intuitionistic reciprocal preference relations, using as a starting point both Zadeh's extension principle and the representation theorem of fuzzy sets. Their findings assist the authors in the building of a novel consistency based IOWA operator capable of associating a higher contribution in the aggregated value to the more consistent information. More recently, Yager and Alajlan [18] addressed the problem of obtaining a consensus subjective probability distribution from the individual opinions of a group of agents; while Wu et al. [13] presented an interesting discussion in the use of uninorm based aggregation methods for group decision-making to propagate trust in the specific context of social networks.

III. IOWA Operators

Yager's Ordered Weighted Averaging (OWA) operator [14] has been proven to be extremely useful in group/multi-criteria decision making problems because it allows to implement the concept of *fuzzy majority* [17]. Recall that an OWA operator of dimension n is a function $F\colon \mathbb{R}^n \longrightarrow \mathbb{R}$ with associated weighting vector $W = (w_1, \ldots, w_n)$ to it, verifying $w_i \in [0,1]$ and $\sum_{i=1}^{n} w_i = 1$, such that

$$F(a_1, \ldots, a_n) = \sum_{i=1}^{n} w_i \cdot a_{\sigma(i)}$$

where $\sigma\colon \{1, \ldots, n\} \longrightarrow \{1, \ldots, n\}$ is a permutation such that $a_{\sigma(i)} \geq a_{\sigma(i+1)},\ \forall i = 1, \ldots, n-1$, i.e., $a_{\sigma(i)}$ is the i-th highest value in the set $\{a_1, \ldots, a_n\}$.

Mitchell and Estrakh in [6] described a modified OWA operator in which the input arguments are not rearranged according to their values but rather using a function of the arguments. Inspired by this work, Yager and Filev in [16] defined a more general type of OWA operator:

Definition 1 (IOWA Operator): An IOWA operator of dimension n is a mapping $I-F\colon (\mathbb{R}\times\mathbb{R})^n \longrightarrow \mathbb{R}$, which has an associated set of weights $W = (w_1, \cdots, w_n)$ to it, so that $w_i \in [0,\, 1]$, $\sum\limits_{i=1}^{n} w_i = 1$,

$$I-F\left(\langle u_1, a_1\rangle, \ldots, \langle u_n, a_n\rangle\right) = \sum_{i=1}^{n} w_i \cdot a_{\sigma(i)}, \qquad (1)$$

and $\sigma\colon \{1, \ldots, n\} \longrightarrow \{1, \ldots, n\}$ is a permutation function such that $u_{\sigma(i)} \geq u_{\sigma(i+1)},\ \forall i = 1, \ldots, n-1$.

In Definition 1, the reordering of the set of values of the argument variable, $\{a_1, \ldots, a_n\}$, is induced by the reordering of the set of values $\{u_1, \ldots, u_n\}$ associated to them, which is based upon their magnitude, and that is referred to as the order inducing vector [16]. An immediate consequence of definition 1 is that if the order inducing vector components coincide with the argument values, then the IOWA operator reduces to the OWA operator. In fact, the OWA operator as well as the weighted average (WA) operator are included in the more general class of IOWA operators, which means that the IOWA operators allow to take control of the aggregation stage of any group/multi-criteria decision making problem in the sense that importance can be given to the magnitude of the values to be aggregated as the OWA operators do or to the information sources as the WA operators do. Thus, the IOWA operator, which generalised the OWA operator, provides an appropriate representation of the *majority concept* in group/multi-criteria decision making processes as it is shown in the following subsections.

A. Linguistic Quantifier Guided OWA Operators

In the same way as other fuzzy logic concepts relate to classical logic, the linguistic quantifier generalises the idea of quantification of classical logic [21, 22]. In classical logic there are two types of quantifiers that can be used in propositions: the universal quantifier (*for all*) and the existential quantifier (*there exists*). According to Pasi and Yager [7], by using linguistic quantifiers we are capable of referencing a variable number of elements of the domain of discourse. This referencing can be done in a *crisp* way or in a *vague* (fuzzy) manner: (a) Crisp: *at least* k of the elements, *half* of the elements, *all* of the elements; (b) Vague (fuzzy): *most* of the elements, *some* of the elements, *approximately* k of the elements.

There are two types of linguistic quantifiers: absolute and proportional. "Absolute quantifiers such as *about* 7, *almost* 6," are be modelled as fuzzy subsets of $\mathbb{R}$, while "proportional quantifiers like *most*, *about* 70%," are modelled as fuzzy subsets of the unit interval, with membership function $Q(x)$ indicating the degree to which the amount/proportion x satisfies the concept Q.

In [14], Yager proposed an approach to give some semantics or meaning to the weights of an OWA operator. This method allowed for applications in the area of quantifier guided aggregations [17] by deriving the weights from a functional form of the linguistic quantifier as follows: let a non-decreasing function Q : $[0,1] \to [0,1]$ such that $Q(0) = 0, Q(1) = 1$ be the fuzzy representation of a proportional monotone quantifier, so that for a given value $x \in [0,1]$, $Q(x)$ is the *degree* to which x *satisfies* the fuzzy concept being represented by the *quantifier*, then the elements of the OWA weighting vector are computed in the following way:

$$w_i = Q\left(\frac{i}{n}\right) - Q\left(\frac{i-1}{n}\right) \tag{2}$$

Examples of linguistic quantifiers are *"at least half"*, *"most of"* and *"as many as possible"*, which can be represented by the following function

$$Q(r) = \begin{cases} 0 & \text{if } 0 \leq r < a \\ \frac{r-a}{b-a} & \text{if } a \leq r \leq b \\ 1 & \text{if } b < r \leq 1 \end{cases} \tag{3}$$

using the values $(0, 0.5)$, $(0.3, 0.8)$ and $(0.5, 1)$ for (a, b), respectively [4].

B. Fuzzy Majority in Collective Decision Making modelled with an IOWA Operator

The semantics behind the aggregation being performed is the key to reflecting the concept of a *majority*. As such, the two alternatives in terms of OWA semantics presented by Pasi and Yager [7] are: (a) OWA operators as an aggregation guided by 'majority' linguistic quantifiers (Subsection III-A), and (b) IOWA operators as drivers of a *majority opinion*. The semantics we propose is (b): using IOWA operators to obtain a *majority opinion* using the monotonic non-decreasing linguistic quantifier *most*.

The final output of an IOWA operator will reflect in a closer manner the opinion of the majority if similar values are closer to each other in the induced vector [7]. This **majority** of values that are similar can be achieved if an IOWA operator has "an inducing ordering variable which is based on a proximity metric over the elements to be aggregated." Thus, what is needed is the ability to calculate the similarities between the values to be aggregated to compute "the values of the inducing variable of the IOWA operator" [7]. Such a function was defined as a binary support function, *Sup*, by Yager in [15] where $Sup_\alpha(a, b)$ expresses the support from b for a at an α level of desired tolerance based on the premise of "the more similar two values are, the more they support each other" [15]:

$$Sup_\alpha(a_i, a_j) = \begin{cases} 1 & \text{if} \quad |a_i - a_j| < \alpha \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

The higher the tolerance α is, the less it is imposed that the two values have to be closer to support each other. If we were to aggregate a set of values and we wanted to order them in increasing order of support, then for each value it is computed the sum of its support values with respect to the rest of values to be aggregated [7]. These overall supports are utilised as the values of the order inducing variable which in turn can be used to induce an ordering based on proximity.

The above is key in utilising IOWA operators to generate a *majority-based aggregation* of the values to be aggregated via a linguistic quantifier Q (*most* in our case). Pasi and Yager's strategy for the construction of the weighting vector from the induce support values is as follows:

1) Include in the definition of the overall support for a_i the similarity of the value a_i with itself:

$$t_i = s_i + 1 \tag{5}$$

2) On the basis of the t_i values, the weights of the weighting vector are computed as follow:

$$w_i = Q\left(\frac{t_i}{n}\right) \Big/ \sum_{j=1}^{n} Q\left(\frac{t_j}{n}\right) \tag{6}$$

"The value $Q(t_i/n)$ denotes the degree to which a given member of the considered set of values represents the *majority*" as per the linguistic quantifier Q.

As such, Eq. (6) is the weights semantic to apply in the proposed *majority* aggregation method.

IV. Cross-ratio Uninorm

Aggregation operators are usually classified into one of the following three categories: (i) *Conjunctive operators* like the family of t-norm operators, which has the minimum operator as its largest element. These operators behave like a logical "and"; (ii) *Disjunctive operators* like the family of t-conorm operators. These operators are the "dual" of conjunctive operators, and they behave like a logical "or". The maximum operator is the smallest of all t-norms operators; and (iii) *Compensative operators* are located between the minimum and the maximum operators, and consequently are neither conjunctive nor disjunctive. These type of operators are known as "averaging operator" and they are widely used in multicriteria decision making problems. The arithmetic mean, the weighted mean, the OWA and IOWA operators are representative examples of this class.

It is worth mention *the family of uninorm operators* [20] as it does not belong fully to any of the three classes described above. Indeed, a uninorm operator, U, is defined as a is a mapping $U\colon [0,1]^2 \longrightarrow [0,1]$ satisfying the properties:

1) Commutativity: $U(x,y) = U(y,x)$
2) Monotonicity: $U(x_1,y_1) \geq U(x_2,y_2)$ if $x_1 \geq x_2$ and $y_1 \geq y_2$
3) Associativity: $U(x, U(y,z)) = U(U(x,y),z)$
4) Identity element: $\exists\, e \in [0,1]: \ \forall\, x \in [0,1], \ U(x,e) = x$

Uninorm, t-norm and t-conorm operators share the commutativity, associativity and monotonicity properties. However, the set of uninorm operators has both the set of t-norm operators and the set of t-conorm operators as its subsets. Indeed, a uninorm operator with "$e = 1$" becomes a t-norm operator; while a uninorm operator with "$e = 0$" becomes a t-conorm operator. In general, a uninorm operator with identity element $e \in]0,1[$ behaves like (i) a t-norm operator when all aggregated values are below e; (ii) a t-conorm operator when all aggregated values are above e; (iii) a compensative operator otherwise.

A. Cross-ratio Aggregative Uninorms

Neither t-norm operator nor t-conorm operators allow "low" values to be compensated by "high" values or viceversa. However, as explained above "uninorm operators may allow values separated by their identity element to be aggregated in a compensating way" [3].

Yager and Rybalov [20] provided the following representation of uninorms in terms of a strictly increasing continuous function of a single variable ϕ: $[0,1] \longrightarrow [-\infty,\infty]$ (generator function):

$$U(x,y) = \phi^{-1}\left[\phi(x)+\phi(y)\right] \;\; \forall x,y \in [0,1]^2 \backslash \{(0,1),(1,0)\}.$$

such that $\phi(0) = -\infty$, $\phi(1) = +\infty$. Chiclana et al. in [2] proved that the and-like representable uninorm operator with $e = 0.5$ and $\phi(x) = \ln\frac{x}{1-x}$ [5], known as the cross-ratio uninorm,

$$U(x,y) = \begin{cases} 0, & (x,y) \in \{(0,1),(1,0)\} \\ \dfrac{xy}{xy+(1-x)(1-y)}, & \text{Otherwise.} \end{cases} \tag{7}$$

is the solution to the functional equation modelling the concept of cardinal consistency of reciprocal preference relations. Fodor [3] extended the cross-ratio uninorm with the identity element $e = 0.5$, so the identity element e can take on any value in $]0,1[$:

$$U(x,y) = \begin{cases} 0, & (x,y) \in \{(0,1),(1,0)\} \\ \dfrac{(1-e)xy}{(1-e)xy+e(1-x)(1-y)}, & \text{Otherwise.} \end{cases} \tag{8}$$

Expressions (8) presents the cross-ratio uninorm as an aggregation operator of two arguments. However, associativity property allows uninorm operators to fuse n (> 2) arguments:

$$U(x_1,\ldots,x_n) = \begin{cases} 0, \\ \quad \text{if } \exists\, i,j : (x_i,x_j) \in \{(0,1),(1,0)\} \\ \dfrac{(1-e)^{n-1}\prod_{i=1}^{n} x_i}{(1-e)^{n-1}\prod_{i=1}^{n} x_i + e^{n-1}\prod_{i=1}^{n}(1-x_i)}, \\ \quad \text{Otherwise.} \end{cases} \tag{9}$$

B. Implementation of Cross-ratio Uninorm in the SA problem

Usually, lexicon-based SA methods are supported by a lexicon containing terms or words capable of conveying sentiments/opinions, most often terms belonging in the part-of-speech categories of adjectives, adverbs, verbs and nouns. Then, when a given document or sentence is analysed, the polarity values of the words/terms present in the text are utilised to calculate the associated semantic orientation, usually positive or negative, and in some cases, neutral.

There are, however, cases in which a lexicon-based method in SA cannot produce a classification outcome. The most common reason for the latter is the absence of terms in the lexicon that are required to process a given sentence. In situations like this, one common approach is to resort to the utilisation of a non lexicon-based approach; typically, a supervised machine learning algorithm. Notice that the semantic orientation discrimination between *positive*, *negative* or *neutral* is in accordance with the behaviour of uninorm operators. Thus, based on the above, we suggest that when a lexicon-based method, like the so-called HSC/HAC technique introduced in [1], is unable to derive the polarity of a sentence then an alternative approach could consist of implementing a uninorm operator, as described in Section IV-A, to aggregate the polarity classification outputs, $\{x_1,x_2,\ldots,x_n\}$, of non-lexicon dependent classification methods, $\{m_1,m_2,\ldots,m_n\}$, respectively. Thus, the resulting aggregation would be defined by $U(x_1,x_2,\ldots,x_n) = \Lambda$, where $\Lambda \in [0,1]$ and U is an appropriate cross-ratio uninorm operator.

Let us assume that the method discussed in [1] is not capable of producing a classification score for a specific sentence. In such a situation, we resort to the outputs of two or more supervised machine learning algorithms, and combine them together using a cross-ratio uninorm as an aggregative operator. This way, a lexicon-based method could utilise this technique as a complement strategy when its lexicon is not in a position to contribute to the generation of a classification outcome. Graphically, the proposed method is depicted in Fig. 1. Notice that this proposed improvement *ensures* that our hybrid classification method (HSC/HAC), as described in [1], will get enhanced as it will always be in a position to produce a classification output.

V. Experimental Results

We have performed a number of tests to validate that the semantics of the proposed operators are indeed what we intended. We have verified the success of these proposed operators using as a context the sentiment analysis (SA) problem at the sentence level, and have used as well the datasets previously put at work in [1].

A. Cross-ratio Uninorm - Results

In Table I, we show comparative results of the cross-ratio uninorm aggregation against two methods: (a) the use of arithmetic mean, and (b) the utilisation of classification outputs attained by using the word-frequency dictionary described in [1], which was utilised as a work-around when the sentiment lexicon did

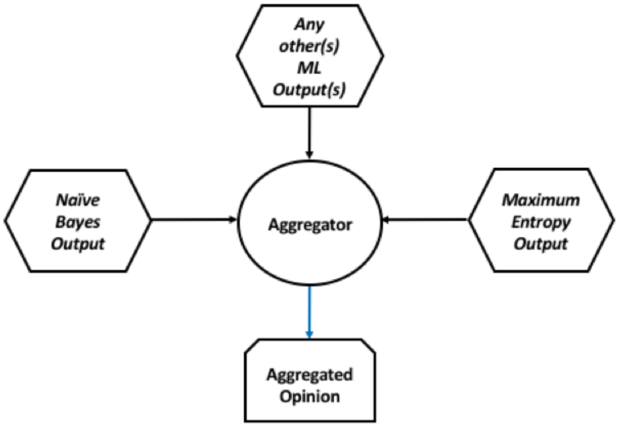

Fig. 1. Cross-ratio Uninorm Aggregation

not count with the required terms (words) when processing sentences. Experiments have been performed for both the Movie Review Dataset (http://www.cs.cornell.edu/people/pabo/movie-review-data/) and the Twitter Dataset *Sentiment140* (http://help.sentiment140.com/for-students).

Method	**Precision - Movie DB**	**Precision - Twitter DB**
Arithmetic Mean	0.5613	0.5723
Word-frequency Dictionary	0.7042	0.6913
Cross-ratio Uninorm	0.8787	0.8811

TABLE I
RESULTS AMONG METHODS - METHOD VS. PRECISION (MOVIE DATASET: 500 SENTENCES; TWITTER DATASET: 50 SENTENCES)

The results are very clear, in the sense that the cross-ratio uninorm **consistently** delivers much better precision than the one exhibited by the two other alternatives used in the comparison process.

B. IOWA Operator driving consensus - Results

We are trying to figure out which method is semantically closer to the opinion of the majority among the participating methods. The first aggregation method used for testing was based on the *Median* Aggregation, whilst the second one was based on the *Arithmetic Mean*. The IOWA operator with the semantic representing the opinion of the majority ($IOWA_{most}$), is incorporated as well in a comparative table, Table II.

Notice that the IOWA operator yields the same results regardless of the tolerance value utilised ($\alpha = 0.3, 0.5$). It still produces a result that is representative of what the majority thinks. What would occur is that the majority agrees on the polarity of a sentence (positive or negative), but depending on the tolerance value α, as presented in the *support function*, Equation 4, a given sentence will be given the same polarity label, but a different intensity value (moderate, very, most, etc.).

Semantic	**Median**	**Arithmetic Mean**	**IOWA$_{most}$** α=0.3,0.5
Opinion of majority	337	388	500
Not opinion of majority	163	112	0
% of success	67.40	77.60	100.00

TABLE II
MEDIAN, ARITHMETIC MEAN & $IOWA_{most}$ AGGREGATIONS

VI. USING THE PROPOSED AGGREGATION METHODS IN ENSEMBLE MODELS

The following is perhaps one of the most accepted definitions of ensemble methods in the field of machine learning. Ensemble methods use multiple learning algorithms to achieve a better performance than the one that could be obtained individually by the constituent methods algorithms alone [10]. A number of options have been used as the ‘combining’ mechanism of ensemble methods. Usually, there are two frameworks available when it comes to ensemble methods: (a) a dependent framework, and (b) the independent framework. In the first one, the output of a given classifier is utilised in the building of the next one, whilst in the case of independent frameworks, the building of all classifiers is based on the same dataset -or subsets of the initial dataset- and the required data is fed to each classifier, but outputs of all classifiers are combined together by an aggregation mechanism [10]. As we can see, the combination (aggregation) method is key in ensemble methods. Among the combining methods, we find weighting approaches (including majority voting, performance weighting, distribution summation, Bayesian combination, order statistics and others) and meta-combination processes (as the likes of stacking, arbiter trees, combiner trees, grading, and others). A complete description of all these combining algorithms can be consulted in [10]. Majority & weighted voting and simple & weighted averaging are some popular combining methods that fall into the categories mentioned above. Given their simplicity in terms of implementation these aforementioned methods are available amongst the most utilised commercial software solutions in the ensemble methods field. Our initial thoughts of finding semantic-sensitive aggregation methods originated from the need of creating an alternative solution for a lexicon-based approach that we introduced in [1]. However, as we proposed the *majority* and *compensative* aggregation operators described in Section III-B and Section IV-A respectively, as solutions to be applied to the SA problem as described in [1], we did realise that these algorithms could be put at work as well as aggregation (combiner) components in ensemble methods. So that, the proposed methods could be utilised specifically in the context of SA, or they could also be deployed, using rather a generalisation approach, into solutions that require aggregating the outputs of any-style classification methods.

VII. Conclusions & Future Work

In this article we have presented two possible techniques associated to specific aggregation semantics - compensative and majority- that could be extremely useful if implemented as the combining element in ensemble classification methods. The obtained results in the SA field have demonstrated that the aggregation methods proposed perform better than other solutions that are commonly used in the commercial software industry. We certainly believe that when one considers the output of several classification methods outputs as the *opinions* of experts, the semantic introduced for the aggregation process to be followed is key in both the final results and their interpretation. As a path forward we will consider exploring and implementing other operators with semantics that could be largely applicable to the industry, among them: (a) the *most* trusted among the experts; (b) the *least* trusted; (c) the values supported by a *by at least half* of the experts; (d) the values supported *by less than half* of the experts; (e) the opinion of the *most prestigious* members of the team of experts.

References

[1] O. Appel, F. Chiclana, J. Carter, and H. Fujita. A hybrid approach to the sentiment analysis problem at the sentence level. Knowledge-Based Systems, 108:110 – 124, 2016.

[2] F. Chiclana, E. Herrera-Viedma, S. Alonso, and F. Herrera. Cardinal consistency of reciprocal preference relations: A characterization of multiplicative transitivity. IEEE Transactions on Fuzzy Systems, 17(1):14–23, 2009.

[3] J. Fodor. On Rational Uninorms. In Proceedings of the First Slovakian-Hungarian Joint Symposium on Applied Machine Intelligence, Herlany, Slovakia, February 12-14, 2003, pages 139–147, 2003.

[4] J. Kacprzyk. Group decision making with a fuzzy linguistic majority. Fuzzy Sets and Systems, 18:105–118, 1986.

[5] E. Klement, R. Mesiar, and E. Pap. On the relationship of associative compensatory operators to triangular norms and conorms. Int. J. Uncertainty, Fuzziness Knoledge-Based Systems, 4:129–144, 1996.

[6] H. B. Mitchell and D. D. Estrakh. A modified owa operator and its use in lossless dpcm image compression. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 5:429–436, 1997.

[7] G. Pasi and R. R. Yager. Modeling the concept of majority opinion in group decision making. Inf. Sci., 176(4):390–414, Feb 2006.

[8] J. Peláez, J. M. Doña, and J. Gómez-Ruiz. Analysis of OWA operators in decision making for modelling the majority concept. Applied Mathematics and Computation, 186:1263–1275, 2007.

[9] J. Perkins. Python Text Processing with NLTK 2.0 Cookbook. Packt Publishing, 2010.

[10] L. Rokach. Ensemble-based classifiers. Artificial Intelligence Review, 33(1):1–39, 2010.

[11] I. J. Rudas, E. Pap, and J. Fodor. Information aggregation in intelligent systems: An application oriented approach. Knowledge-Based Systems, 38:3–13, Jan 2013.

[12] J. Wu and F. Chiclana. Multiplicative consistency of intuitionistic reciprocal preference relations and its application to missing values estimation and consensus building. Knowledge-Based Systems, 71:187–200, Nov 2014.

[13] J. Wu, R. Xiong, and F. Chiclana. Uninorm trust propagation and aggregation methods for group decision making in social network with four tuple information. Knowledge-Based Systems, 96:29–39, 2016.

[14] R. Yager. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decisionmaking. Systems, Man and Cybernetics, IEEE Transactions on, 18(1):183–190, Jan 1988.

[15] R. Yager. The power average operator. Systems, Man, and Cybernetics, Part A: Cybernetics, IEEE Transactions on, 31:724–730, 2001.

[16] R. Yager and D. Filev. Induced ordered weighted averaging operators. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 29(2):141–150, Apr 1999.

[17] R. R. Yager. Quantifier guided aggregation using OWA operators. International Journal of Intelligent Systems, 11(1):49–73, 1996.

[18] R. R. Yager and N. Alajlan. An intelligent interactive approach to group aggregation of subjective probabilities. Knowledge-Based Systems, 83:170–175, July 2015.

[19] R. R. Yager and N. Alajlan. Some issues on the owa aggregation with importance weighted arguments. Knowledge-Based Systems, 100:89–96, May 2016.

[20] R. R. Yager and A. Rybalov. Uninorm aggregation operators. Fuzzy Sets and Systems, 80(1):111 – 120, 1996. Fuzzy Modeling.

[21] L. A. Zadeh. The concept of a linguistic variable and its applications to approximate reasoning - I. Information Sciences, 8:199–249, 1975.

[22] L. Zadeh. A computational approach to fuzzy qualifiers in natural languages. Computing and Mathematics with Applications, 9(1):149–184, 1983.