International Journal of Computational Intelligence Systems
Vol. **13(1)**, 2020, *pp.* 56–65
DOI: https://doi.org/10.2991/ijcis.d.200121.001; ISSN: 1875-6891; eISSN: 1875-6883
https://www.atlantis-press.com/journals/ijcis/

Coreference Resolution Using Neural MCDM and Fuzzy Weighting Technique

Samira Hourali, Morteza Zahedi*, Mansour Fateh

School of Computer and IT Engineering, Shahrood University of Technology, Shahrood, Iran

ARTICLE INFO

Article History

Received 08 Nov 2019
Accepted 13 Jan 2020

Keywords

Coreference resolution
Fuzzy weighting
Text mining
Mention extraction
Kohonen neural network

ABSTRACT

Coreference resolution has been an active field of research in the past several decades and plays a vital role in many areas such as information extraction, document summarization, machine translation, and question answering systems. This paper presents a new coreference resolution approach by incorporating RoBERTa embedding with a neural multi-criteria decision making (MCDM) method. The proposed model does not use any syntactic and dependency parser. Mentions were extracted from the text with an unhand engineered mention detector and features were extracted from a deep neural network. Next, the problem is modeled in the form of effective parameters of the performance such as error rate reduction and enhances the F1 by Kohonen MCDM neural network. The weights assigned to the features represent their importance and suggests the best reference for a mention where such weights are computed using a fuzzy weighting method. Comparing to state-of-the-art coreference resolution models, the simulation results show significant improvements for the proposed approach on different datasets in terms of precision and recall and achieving marginal improvements on the following datasets: English CoNLL-2012 shared task (+3.1 F1), Yahoo's news site (+6.6 F1), and English Gigaword (+7.04).

1. INTRODUCTION

The process of finding co-referent mentions (mentions that refer to the same entity in real-world) in a document is called coreference resolution which is considered as one of the most important challenges in the field of text processing. The process humans use to identify co-referent mentions in conversations or texts is still not clear. Also, it is still difficult to examine the knowledge of this procedure. Hence, coreference resolution, an effective process in dealing with subjects such as information extraction, machine translation, text summarization, and Q & A systems, is considered as an important issue in the field of natural text and language processing. Coreference resolution has been an active research topic over the past four decades, but its complete solution has not yet been presented. The presented solutions have at least three important drawbacks. Firstly, the accurate computational model is not provided to solve this problem. New approaches [1–3] for coreference resolution generally use deep learning and reinforcement learning, but they did not present an accurate computational model. Secondary, most of the coreference resolution problems can only be resolved using various knowledge resources including lexical knowledge, syntactic knowledge, world knowledge, and semantic knowledge. Currently, most of the coreference resolution systems [3–6] are not equipped with these knowledge resources. Thirdly, most of the machine learning methods in this field has been done in the English language and apply them to other languages leads to challenging in the time-consuming task of creating annotation documents.

In the proposed approach, we tried to improve the accuracy of the coreference resolution by extracting better features and providing a better architecture than previous approaches [1–3,7]. For this purpose, the RoBERTa [8] method has been used for considering the syntactic and semantic knowledge and extracting correct mentions with different length from the text. Better contextual information compared to existing works is provided to the mention detection system using this idea. Then, candidate antecedents are ranked accurately for the intended mention by multi-criteria decision making (MCDM) structure, based on the Kohonen neural network. MCDM is designed to deal-with problems of different number of choices. This approach includes multiple stages such as, identifying the goal of the decision-making process, selection criteria, selection of alternatives, selection of the weighing methods, and aggregation [9]. First step involves the correct identification of the goal or the final output of the decision-making process. In the second step, independent and consistent criteria are selected which should have a miserable and similar scale and should be inter related with the alternatives. In the third step, available and comparable alternatives were selected. In the fourth step, the importance of each criteria is identify which can be determined using weighting methods. In final step, the best alternative is selected from available options by desired ranking method. The neural MCDM model is an accurate computational model that improve F1 on the test set of the English CoNLL-2012 shared task by 3.1. On one hand, using

*Corresponding author. Email: zahedi@shahroodut.ac.ir

the neural network for decision making has the advantage of parallel execution. Therefore, it has a great effect on reducing execution time. On the other hand, considering all features and their degree of importance lead to accurate ranking alternatives. Also, we directly consider all spans in a document as potential mentions and the mention detection accuracy is improved in comparison to the newest method of mention extraction [1]. The rest of the paper is structured as follows. The related research is provided in Section 2. In Section 3, the proposed model is presented. In Section 4, the experimental results of this study are presented considering CoNLL-2012 [10], MUC6 [11], and English Gigaword [12] datasets. Finally, the conclusion is drawn in Section 5.

2. RELATED WORKS

Coreference resolution models generally divided into three categories [13]: (1) mention-pair model, mention-ranking model, entity-level model. The mention-pair model operates based on a pair of mentions in which, two mentions are either co-referent or not; then, by combining all the co-referent mentions, the coreference chains are identified in the document [14–16]. (2) In the mention-ranking model, instead of exploring whether two mentions are co-referent or not, by searching among a group of mentions, the best candidate for desirable mention is found [5,17,18]. (3) The entity-level model categorizes each mention with the previous entity instead of categorizing each mention with the previous mention. As a result, it creates a collection of mentions for each entity. This approach usually uses clustering methods [15,16,19,20].

In general, coreference resolution methods are divided into rule-based methods, machine learning-based (statistical), and deep learning-based groups. In rule-based methods [21–28], a collection of rules are handwritten by experts. These rules are implemented in an orderly manner to specify co-referents in the text. One of the advantages of the rule-based method is a high level of accuracy and simplicity in design. However, this method has low flexibility so that experts should register the system from scratch for any natural language. Machine learning methods are also classified into supervised and unsupervised categories. The former, use the educational data for learning the system that must initially be written by individuals [15,20,29,30]. On the other hand, no educational data (or at least very little data) are required in unsupervised methods [31,32]. However, the accuracy level of this method is currently too low to solve the coreference resolution problem.

New methods of coreference resolution [1,4,5,7,17,18] use deep neural networks. One of the main advantages of using deep neural networks is the use of raw text in the input of the model so that useful features are extracted from the text within the network itself and without human intervention. In addition, these methods use vectors (word embedding) to represent words and describe semantic relationships between them. Although more recent deep learning works in this area have used cluster-level information and such information is necessary to prevent the connection of non-co-referent mentions and the formation of co-referent chains, these methods have still a lot of shortcomings such as inappropriate cost functions, great dimensions, and inappropriate architecture. Accordingly, the proposed approach has tried solving these problems by using contextual, semantic and syntactic information for better representation of spans and neural MCDM method for accurate ranking candidate antecedent and better detection rate of co-referent mentions. Peng *et al.* [33] proposed an emergency decision-making approach based on a weighted distance-based approximation (WDBA) method which can obtain the optimal alternative without counterintuitive phenomena and possess a strong ability to differentiate the optimal alternative. They also proffered a new score function for q-rung orthopair fuzzy number (q-ROFN), which takes the hesitation information into consideration that reduce the information losses. After this idea, Peng *et al.* [34] proposed a new MCDM method by means of the q-rung orthopair fuzzy weighted exponential aggregation (q-ROFWEA) operator and q-rung orthopair fuzzy exponential weights which have a wide range to describe the real case. They also proffered a new score function of q-rung orthopair fuzzy number (q-ROFN) for solving the failure problems when comparing two q-ROFNs. In this paper we proposed MCDM method based on the Kohonen neural network where the weights of criteria are calculated by the fuzzy method and neural structure for decision making leads to we can accurately find appropriate candidate antecedent for the desired mention.

3. PROPOSED MODEL

In this section, the proposed model, along with the building formulation process is discussed. The proposed model consists of two parts. In the first part, mentions are extracted from the word sequence by pre-trained language models and deep neural networks. Then, in the second part, co-referent mentions are identified by the MCDM method.

3.1. Mention Detection

The first phase in coreference resolution is mention detection. In this phase, mentions in the text such as named entities [35], noun phrases and pronouns are identified and extracted. This is done through the following steps (token representation, span representation, and mention scoring). Each word or token in the input sentence is a combination of two vectors (character embedding and word embedding) to preserve semantic and syntactic information. RoBERTa [8] was used for embedding words because this method uses a pre-trained language model and considers the probabilities of a sequence of words. RoBERTa is designed to pre-train deep bidirectional representations from the unlabeled text by jointly conditioning on both left and right context in all layers. We fine-tune the pre-trained RoBERTa model to desire world knowledge in model. Consider the input document D contains T word. Assume vector representation of each word is $\{x_1, \ldots, x_T\}$. For computing vector representation of each span, we fed word representations to the bidirectional gated recurrent unit (GRU) according to Eqs. (1–5) and encode every word in the context of the span. A GRU at position t has two gates, an update gate z_t, and a reset gate r_t. More specifically, each GRU can be expressed as follows:

$$r_t = \sigma\left(W_{rx}x_t + W_{rh}h_{t-1}\right) \tag{1}$$

$$z_t = \sigma\left(W_{zx}x_t + W_{zh}h_{t-1}\right) \tag{2}$$

$$\tilde{h}_t = tanh\left(W_{hx}x_t + W_{hh}\left(r_t \odot h_{t-1}\right)\right) \tag{3}$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t, \tag{4}$$

$$x_t^* = [h_{t+1}, h_{t-1}] \tag{5}$$

In Eq. (1) W's are model parameters of each unit; $\tilde{h}_t$ is a candidate hidden state which is used to compute h_t; σ is an element-wise sigmoid logistic function defined as $\sigma(x) = 1/(1 + e^{-x})$; $\sigma \epsilon [-1, 1]$ indicates the directionally of each GRU and $\odot$ denotes element-wise multiplication of two vectors; x_t^* is the concatenated output of the bidirectional GRU. Independent GRUs are used for every sentence. Previous systems [5,17,20] typically used syntactic heads as features. Our model, however, uses multi-step head-finding attention [36] to compute a score distribution over different words in a span s_i and learns the task-specific notion of headedness. This method uses a separate attention mechanism for each decoder layer. Weighted embedding of each span is computed according to Eqs. (6–8), where $\hat{x}_i^l$ is a weighted sum of word vectors in span i; g_i is the embedding of the previous target words; L shows the layer of multilayer FFNN; b_α^l is the bias parameter; e_i is the embedding of the input word. Here, the weights $a_{i,t}^l$ are automatically learned. We encode structure and contextual information of spans to effective span representations. The structure of the proposed model is shown in Figure 1.

$$\alpha_t^l = w_\alpha^l . FFNN_\alpha \left(x_t^*\right) + b_\alpha^l + g_i \tag{6}$$

$$a_{i,t}^l = \frac{exp\left(\alpha_i^l . x_t^u\right)}{\sum_{t=START(i)}^{END(i)} exp\left(\alpha_i^l . x_k^u\right)} \tag{7}$$

$$\hat{x}_i^l = \sum_{t=START(i)}^{END(i)} a_{i,t}^l \left(x_t^u + e_i\right) \tag{8}$$

We concatenate the above span information. This span information contains a feature vector f(i), headword vector $\hat{x}_i^l$, and span size which includes boundary representations to produce the final span representation s_i for span i. Finally, the span representation is obtained by Eq. (9).

$$s_i = \left[x_{START(i)}^*, x_{END(i)}^*, f(i), \hat{x}_i^l\right] \tag{9}$$

After extracting vector representations s_i for each possible span i, according to Eq. (10) we fed them as input to the feed-forward neural network to determine whether they are mention or not and span received mention score.

$$s_m(i) = w_m . FFNN_m(s_i) \tag{10}$$

3.2. Detecting Co-referent Mentions

After detecting mentions, to detect co-referent mentions we formulate the problem as MCDM problem. In this idea, alternatives are candidate antecedents for each mention and features are criteria. We use the following stages to detect co-referent mentions.

3.2.1. Decision matrix

To make a decision, the problem should be formulated using a matrix. For each mention m, we consider 50 candidate antecedents

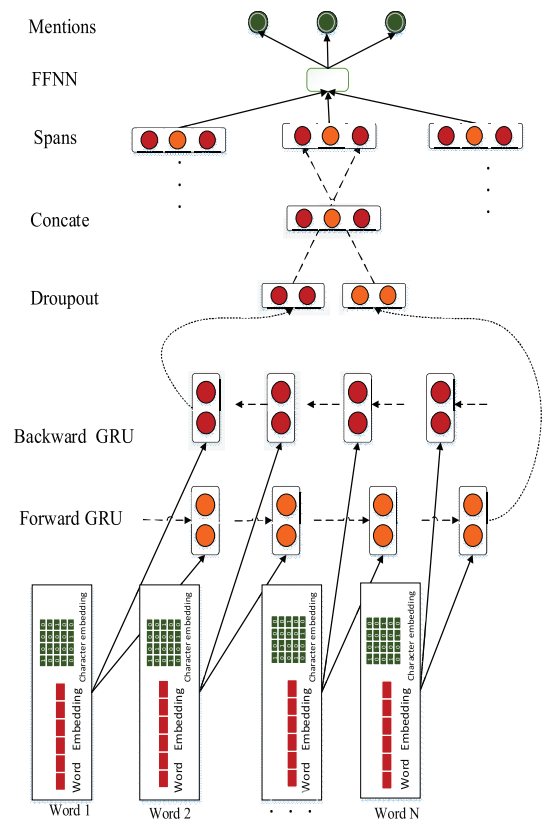

Figure 1 Structure of the proposed model.

and construct a decision matrix M. This matrix is displayed in Eq. (11).

$$M = \begin{array}{c} \\ c_1 \\ c_2 \\ \dots \\ c_m \end{array} \begin{array}{c} \begin{array}{cccc} f_1 & f_2 & \dots & f_n \end{array} \\ \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \dots & \dots & \dots & \dots \\ r_{m1} & r_{m2} & \dots & r_{mn} \end{bmatrix} \end{array} \tag{11}$$

In Eq. (11), c_i is an indicator of the candidate antecedent i; f_j denotes the j-th feature and r_{ij} represents the value of j feature for the i-th mention.

3.2.2. Scaleness of the matrix P

The measurement scales of the quantitative features can be different. For this reason, performing basic math operations before scaling or equalizing the scales is not allowed. Therefore, considering Eq. (12), every r_{ij} element from the assumed decision matrix is divided into the softness of the j column (for the index f_j).

$$x_{ij} = \frac{r_{ij}}{\sum_{i=1}^{n} r_{ij}} \tag{12}$$

In this way, all columns of the matrix have an equal length unit and, therefore, their overall comparison will be easy.

3.2.3. Calculation of the weight of features

The geometric mean of each row and the weight of the i-th feature are calculated by Eq. (13) [37].

$$z_i = \left[\prod_{j=1}^{n} x_{ij}\right]^{\frac{1}{n}} \tag{13}$$

An indicator of the weight and importance of the i's feature for every mention in the text is obtained by Eq. (14).

$$w_i' = \frac{z_i}{(z_1 + z_2 + \ldots + z_n)'}, \forall_i \tag{14}$$

Considering Eq. (14), to generalize the above-mentioned method into the fuzzy state, the classic operators must be replaced by other operators like fuzzy sum, fuzzy multiplication, converting numbers to trapezoid fuzzy numbers, etc. Therefore, steps A, B, and C should be followed:

A. In this step, the paired matrix, the elements of which are trapezoid fuzzy numbers, is identified by the decision-maker. If the preference of the i-th element is shown as $\tilde{a}_{ij} = (a_{ij}, b_{ij}, c_{ij}, d_{ij})$, the preference of the i-th element would be as $\tilde{a}_{ij} = \left(\frac{1}{a_{ij}}, \frac{1}{b_{ij}}, \frac{1}{c_{ij}}, \frac{1}{d_{ij}}\right)$, assuming i = j, we can say $\tilde{a}_{ij} = (1, 1, 1, 1)$.

B. To calculate the features' weight with the fuzzy technique, the geometric mean of each row of comparison matrices is calculated by Eq. (15).

$$\tilde{z}_i = (\tilde{a}_{i1}, \tilde{a}_{i2}, \ldots, \tilde{a}_{in})^{\frac{1}{n}} \tag{15}$$

Then, the fuzzy weight is obtained by Eq. (16) [37,38].

$$\tilde{w}_i = \tilde{z}_i . (\tilde{z}_1 \oplus \tilde{z}_2 \oplus, \ldots, \tilde{z}_n \oplus)^{\frac{1}{n}} \tag{16}$$

C. The final weight of features is calculated by considering the combined method for each mention.

In this step, the weights are assigned to features by considering the values of feature for each mention and candidate mentions in the text. The total weight is considered to be one. It should be mentioned that relative importance (weight) is an indicator of the feature's priority in the decision-making process. These weights are calculated by Eq. (17) and by the integration of the vector $m = [r_1 \ r_2 \ \ldots \ r_n]$ with the vector $\tilde{w}$:

$$w_i = \frac{r_i \tilde{w}_i}{\sum_{i=1}^{n} r_i \tilde{w}_i}, \sum w_i = 1 \tag{17}$$

3.2.4. Weighting the decision matrix

To consider the weight of features in the decision matrix, it is necessary to multiply each column of the matrix by the calculated weights according to Eq. (18).

$$V = \begin{bmatrix} w_1 x_{11} & w_2 x_{12} & \ldots & w_n x_{1n} \\ w_1 x_{21} & w_2 x_{22} & \ldots & w_n x_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ w_1 x_{m1} & w_2 x_{m2} & \ldots & w_2 x_{mn} \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} & \ldots & v_{1n} \\ v_{21} & v_{22} & \ldots & v_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ v_{m1} & v_{m2} & \ldots & v_{mn} \end{bmatrix} \tag{18}$$

3.2.5. Normalization of the matrix M

Weighting the decision matrix makes the matrix droplets have small values. For a matrix to be considered as a neural network input, a normalization step is required. The relationship is used to normalize the weighted matrix.

$$a_{ij} = \frac{v_{ij}}{\max_j \{v_{ij}\}} \tag{19}$$

3.2.6. The final ranking of candidate antecedents by Kohonen neural networks

The network used in the proposed approach has n input and a fixed number of neurons in the output layer. The number of output layer neurons can be considered constant or variables. The number of samples in the training set is derived from the number of output neurons. Examples of training are representative of the alternatives and should, therefore, be selected to cover all the possible situations. As the input matrix to the network is normal, the values of the instruction samples should be such that the interval [1 and 0] is fully covered.

T_i samples are introduced for network education. The output of the Kohonen network [39] will be the neuron that has the most similar or least Euclidean distance with the input sample. As the number of output neurons for each problem with any number of alternatives is considered constant, the actual output values of the network are analyzed. The value of the training parameter and the neighborhood function are:

$$\eta(t) = 1 \tag{20}$$

$$N(t) = \begin{cases} \frac{x}{\alpha} + 1 & x > \alpha \\ -\frac{x}{\alpha} + 1 & x < \alpha \\ 0 & otherwise \end{cases} \tag{21}$$

The neighboring function is shown in Figure 2. The value of α changes depending on the training sample. The values for

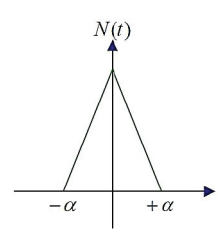

Figure 2 Neighborhood function employed in the Kohenon network.

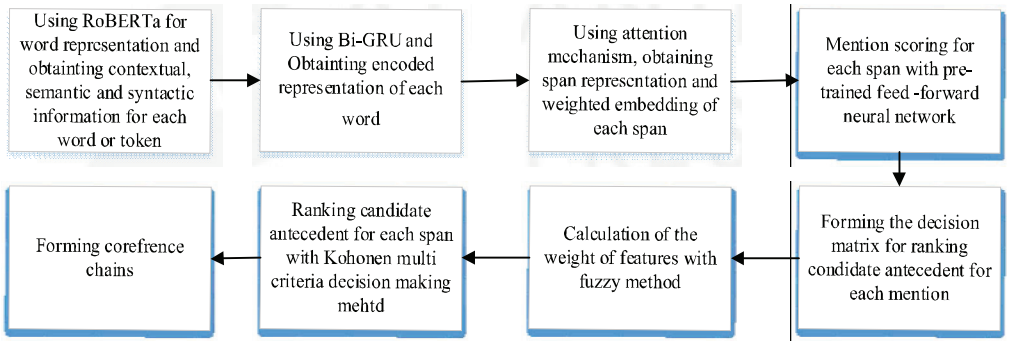

Figure 3 | Block diagram of proposed approach.

each alternative of decision-making matrix are introduced to the network, and according to the network output, each of the alternatives belongs to one of the sets related to the output. In this way, general sorting is performed. To sort the alternatives, the true output values of the network are analyzed. The decision-making indicator at this stage of sorting is considered as follows:

$$b_i = p^i_{j-1} - p^i_{j+1} \tag{22}$$

In Eq. (22), p^i_{j-1} and p^i_{j+1} respectively are the Euclidean distances of the input sample with the previous neuron and the next neuron corresponding to the output of the network. The b_i parameter is used to sort the alternatives in each class. We will have in each class:

$$C_i > C_j \qquad if \qquad b_i > b_j \tag{23}$$

In this way, candidate antecedents for the considered mention based on their features are ranked. In this way, candidate antecedents with a higher rank (highest rank is 1) are more closely related and more similar to the intended mention, and the probability of their coreferention is higher. The block diagram of proposed approach is shown in Figure 3.

4. EXPERIMENTAL RESULTS

4.1. Datastes

To evaluate the effectiveness of the proposed approach, we use four datasets in our experiments. The first dataset is CoNLL-2012 [10] shared task which is a standard coreference resolution corpus for multilingual languages (English, Chinese, and Arabic). We use the English coreference resolution data from the CoNLL-2012 shared task in our experiments. This dataset contains 2802 training documents, 343 development documents, and 348 test documents. The training documents contain on average 454 words and a maximum of 4009 words. The second dataset is the (Message Understanding Conference) MUC6 [11] which was produced by Linguistic Data Consortium (LDC) and contains 318 annotated Wall Street Journal articles, the scoring software and the corresponding documentation used in the MUC6 evaluation. The third dataset is English Gigaword [12] which is intended to evaluate the power of identifying the named entities. This dataset is a comprehensive archive of English news articles and has been used at Pennsylvania University for many years. The fourth dataset is Yahoo's news site which is manually annotated and contain 100 news document.

4.2. Implementation and Hyperparameters

We extend the original Tensorflow implementations of RoBERTa. We fine-tune all models on the CoNLL-2012 English data for 19 epochs using a dropout of 0.4, and learning rates of 1×10^{-4} and 2×10^{-3} with linear decay for the RoBERTa parameters and the task parameters respectively. We found that this made a sizable impact of 2%–3% overusing the same learning rate for all parameters. The hidden states in the GRUs have 200 dimensions. Each feedforward neural network consists of two hidden layers with 150 dimensions and rectified linear units.

4.3. Results

Average F1 in comparison with other methods: Table 1 compares the results of our system with state-of-the-art approaches [1,3–7]. The reported results are either adopted from their papers or reproduced from their code.[1] The comparison is on the CoNLL-2012 test set, according to MUC (mention based) [40], B^3 [41] (link-based), CEAFφ_4 [41] (optimal mapping based) metrics and values of precision, recall, and average F1 score (CoNLL F1 or average F1 of MUC, B^3 and CEAFφ_4). The main evaluation is the average F1 of the three metrics. As shown in Table 1, our model improves the state-of-the-art average F1 by 3.1. Thus, the proposed approach improves recall due to the inclusion of mentions which were otherwise ignored in recent researches. Other methods ignore the missing mentions in the process of identifying them. Here, though, only mentions with more than 10 words were ignored, which account for less than 1.9% of all the mentions. Moreover, due to the correct identification of entities, improvement in precision and recall values for CEAFφ_4 is greater than those of other methods and entity-level information prevents incorrect merging of the clusters. Also, by considering all the features in the mention detection phase, better scores were produced for mentions and according to Kohonen network, co-referent mentions are better recognized.

To better investigate the process of ranking candidate antecedents for mentions, we also compared Kohonen and perceptron networks as follows:

Candidate antecedents ranking with perceptron network: To better investigate the process of ranking candidate antecedents for

[1] https://github.com/kkjawz/coref-ee, https://github.com/kentonl/e2e-coref, https://github.com/clarkkev/deep-coref.

Table 1 Results on the test set on the English data from the CoNLL-2012 shared task (test set).

Method	MUC			B^3			CEAFφ_4			
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Avg.F1
[5]	73.6	65.6	69.4	67.4	56.9	61.7	62.4	58.6	60.4	63.8
[4]	81.2	73.6	77.2	72.3	61.7	66.6	65.2	60.2	62.6	68.8
[6]	79.4	73.8	76.5	69.0	62.3	65.5	64.9	58.3	61.4	67.8
[1]	81.4	79.5	80.4	72.2	69.5	70.8	68.2	67.1	67.6	73.0
[7]	85.4	77.9	81.4	77.9	66.4	77.7	70.6	66.3	68.4	73.8
[3]	82.6	83.4	83.0	73.3	76.1	74.7	72.3	71.1	71.7	76.6
[2]	84.7	82.4	83.5	76.5	74.0	75.3	74.1	69.8	71.9	76.9
Our method	84.1	88.2	86.1	77.5	80.9	79.2	74.3	75.3	74.8	80.0

mentions, we also trained a perceptron network for decision making and compared its performance with that of the Kohonen. In order to use the perceptron network [42], it is essential to identify a series of input samples and their corresponding outputs for training. In supervised networks, the main issue is to determine the number of samples for training. In order to solve decision problems, the samples of the training set should be so that all the different modes of alternatives are included. The perceptron network outputs correspond to the sorted list of alternatives, so the number of neurons in the network output layer is equal to the number of states where the alternatives are relative to each other. The proposed network includes m alternative and n criterion, a network with $m \times n$ input and $m!$ output. The learning parameter is set to $\eta = 1$. The output function is competitive so the neuron with the highest value would be set to one while the rest change to zero. The output function $\varnothing(v(t))$ is given by Eq. (24). Note that the network weights were initialized to a vector of zeros.

$$\varnothing(v(t)) = \begin{cases} 1 : max(v(t)) \\ 0 : otherwise \end{cases} \tag{24}$$

Samples of training were obtained according to Eqs. (25–28).

$$f^j_{min} = \min_i a_{ij} \tag{25}$$

$$f^j_{max} = \max_i a_{ij} \tag{26}$$

$$T_i = \begin{bmatrix} t_i^{11} & t_i^{12} & \dots & t_i^{1n} \\ t_i^{21} & t_i^{22} & \dots & t_i^{2n} \\ \dots & \dots & \dots & \dots \\ t_i^{m1} & t_i^{m2} & \dots & t_i^{mn} \end{bmatrix} \tag{27}$$

$$t_i^{jk} = \frac{f^j_{max} - f^j_{min}}{m-1}(j-1) + f^j_{min} \tag{28}$$

By relocating the rows of the T_i matrix, other samples were obtained. The number of states that m alternatives can have about each other is equal to $m!$. Therefore, relocating m different rows of the T_i matrix can produce all possible combinations for the training network. These samples are introduced with the corresponding output to train the network. After the training, the decision matrix will be introduced to the network and according to the network output, the priority of the alternatives will be determined (that is, relative to each other), and thereafter, the candidate mentions (alternatives) will be ranked according to the desired mention.

Comparison of Kohonen and perceptron network for mention ranking: using the neural network for ranking candidate antecedents has the advantage of parallel execution. Therefore, it has a great effect on reducing computation. As shown in Table 2, the use of a perceptron network has increased run time and F1 on CoNLL-2012 development set to 76.7. When using supervised networks such as the perceptron network, the main problem is determining samples of the training set. The results of these types of networks are dependent on the variety of training samples. If the training samples are selected with enough diversity, the network can predict correctly in response to unseen data. In order to provide such examples of training in supervised networks, the rule of thumb is the more information we use, the better the results will be. Note that due to the intrinsic nature of unsupervised networks, such as that of the Kohonen, the characteristics of such networks are independent of the problem, and a fixed number of neurons should be considered in the output layer. Also, fixed samples are used to train the network. This is independent of the problem and does not depend on the number of alternatives. Therefore, a trained network can be used several times for different problems.

Mention detection rate: As described in Section 3.1, we used bidirectional GRU for extracting spans from sentence and mention detection. We also used Recurrent neural network (RNN), long short-term memory network (LSTM), Bidirectional LSTM and GRU for this purpose. Table 3 compares the strength points of all RNN-based networks. as can be seen, bidirectional-GRU has a better performance than other types of RNNs, as GRU networks have better performance in long dependency modeling and clearly outperform simple RNNs. In addition, novel GRU outperform LSTM networks. We suggest that the reason for such superiority is that GRUs combine the forget and input gates into one update gate which makes it faster to compute. Moreover, bidirectional GPUs improve the performance of RNN in dependency modeling. Therefore, by using RoBERTa word embedding as the input of the network and considering the semantic information of words; mention detection and coreference resolution process are done significantly better than the previous methods.

Figure 4 compares the accuracy of detect mentions from spans in word sequences with that of other methods. as can be seen, in previous methods, the recognition accuracy of the mentions is significantly reduced with an increase in the length of the word sequence. However, in the proposed approach, this reduction is small. Also, for spans with more than five words, the accuracy reduction is negligible. The important advantage of the proposed model is the ability to detect unknown mentions which are not in the training set. As outlined in Liang and Wu [21], there is a large

Table 2 F1 reduction with delete or change the features, word embeddings, and ranking methods on the CoNLL-2012 development set.

	Avg.F1	Δ
– Our model (Kohonen network)	79.9	
– Our model (perceptron network)	76.7	−3.2
– Glove	77	−2.9
– ELMo	75.9	−4
– RoBERTa	75.2	−4.7
– Distance and width features	75.6	−4.3
– Speaker and genre metadata	76.6	−3.3
– String structure matching	76.3	−3.6

Table 3 Mention detection power in all types of RNN-based networks.

	Prec.	Rec.	F1
RNN	79.21	83.12	81.11
LSTM	81.39	85.98	83.62
Bidirectional LSTM	84.19	90.81	87.37
GRU	85.32	93.91	89.40
Bidirectional GRU	88.93	97.95	93.22

overlap between gold mentions and the development set. The proposed model can correctly identify 1059 mentions (394 mentions in the training set and 665 mentions that had not been seen in the training set) which were not recognized previously such as Lee *et al.* approach [1].

Named entity detection rate: As mentioned in the mention detection section, named entities are also a part of the mentions in the text. In Figures 5 and 6 we evaluate and compare the performance of the proposed approach in terms of identifying named entities with the available methods in this field (Stanford-NLP,[2] OpenNLP,[3] LingPipe,[4] SupersenseTagger,[5] AFNER,[6] AlchemyAPI[7]) on the English Gigaword dataset. The proposed approach uses a pre-trained neural network structure to identify the mentions, and this type of network is capable of automatically extracting features from entities and thus, this method has a better performance than all the ones compared. The precision and recall values for named entity detection rate were computed respectively by Eqs. (29) and (30).

$$Precision = \frac{Total\ number\ of\ true\ extracted\ entities}{Total\ number\ of\ extracted\ entities} \tag{29}$$

$$Recall = \frac{Total\ number\ of\ true\ extracted\ entities}{Total\ number\ of\ true\ entities} \tag{30}$$

As shown in Figure 6, the Avg.F1 for the proposed approach has improved up to 7.04, compared with previous methods. This is due to using RoBERTa method which provides morphological information and a solution to backoff for out of vocabulary words. The proposed approach can also identify rare named entities. Therefore,

[2] http://nlp.stanford.edu/.

[3] http://opennlp.apache.org/.

[4] http://alias-i.com/lingpipe/.

[5] http://sites.google.com/site/massiciara/.

[6] http://afner.sourceforge.net/afner.html.

[7] http://www.alchemyapi.com/.

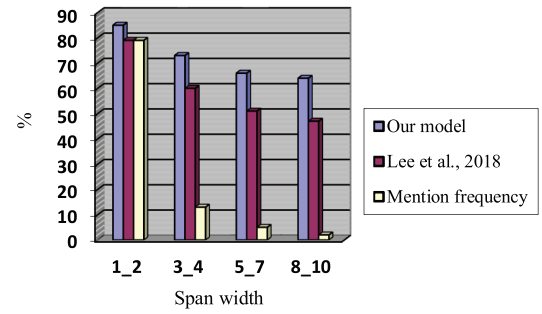

Figure 4 Mention detection rate based on span width in comparison with Lee [1] model.

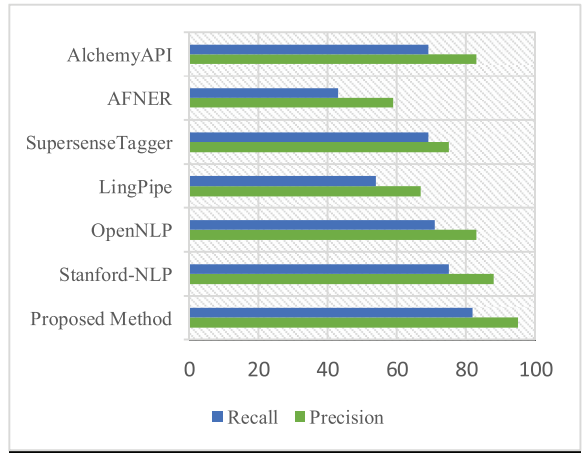

Figure 5 Precision and recall comparison for named entity recognition.

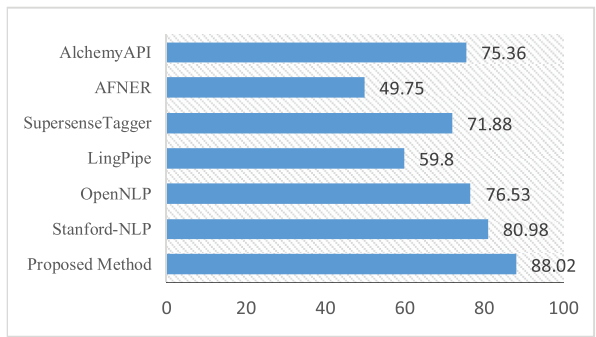

Figure 6 F1 comparison for named entity recognition.

the combination of these two methods can be a suitable approach to identifying entities, in which the precision and readability values for each entity (such as persons, locations, and organizations) should be calculated separately.

Ablations: To show the importance of each component in our proposed model, we ablate various parts of the architecture and report the average F1 on the development set of the data.

Word embedding: In Table 2 GloVe [43], ELMo [44], and RoBERTa embedding methods have been compared in terms of impact on average F1. As the results show, the ELMo has a greater improvement on the F1 value, due to deep contextualized word representation, consideration of syntactic and semantic characteristics of words, and use of pre-trained language models. RoBERTa also

Table 4 Comparison proposed model with based coreference resolution systems on Yahoo's news site.

	MUC			B^3			CEAFφ_4			
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Avg.F1
Our model	81.0	73.8	77.2	70.3	63.7	66.6	68.6	61.8	65.0	69.6
Stanford system [14]	76.1	69.4	72.6	65.6	56.0	60.4	59.4	53.0	56.0	63.0
Illinois system [45]	52.3	57.2	54.6	63.6	52.5	57.5	54.3	54.4	54.3	55.4

performs better than both of these methods. As a result, in our approach, more useful semantic information is available to the system.

Features: The effect of the deletion of features from the proposed system is also reviewed. Eliminating the essential features of the coreference resolution, such as the distance between the spans and the lengths of spans has more effect on the reduction of F1 than the removal of the string structure matching feature. Also, performance degrades by 3.3 F1 without speaker and genre features.

Comparison with based coreference resolution systems: In Table 4, the proposed approach is compared with two base coreference resolution systems, Illinois [45] and Stanford [15] on Yahoo's news site. as can be seen, the results show significant improvement than base systems. Illinois system is only well-suited to the English language. This reason is that similar features are used for all languages, and hence, this system does not properly function for some languages like Chinese.

The reason for choosing strange news is the existence of a large number of events and verbs. Therefore, the proposed approach should best be evaluated using a large number of events. The proposed approach was highly efficient in evaluating the textual documents of the second test set for two major reasons. First, these documents contain a large number of events and propositions and that means more connections between the arguments' mentions of these propositions. Second, various features and aspects of a mention are usually used in these texts that refer to that particular mention.

For example, to understand the content of an accident caused by a person, some words and phrases such as the name of the person, the role he played in the incident, his age, etc. are used. Therefore, common features, used for identifying co-referent mentions in coreference systems were unable to identify these co-referent cases. As a result, using the decision-making and weighting system can create a semantic knowledge in the processing system. In this way, the proposed approach can correctly identify more cases of co-referent mentions, compared to those coreference systems lacking such knowledge.

5. CONCLUSION

In this paper, a coreference resolution method, based on deep learning and fuzzy weighting method, was proposed. In this algorithm, a proper reference was chosen for the mention considering all the extracted features. MCDM for ranking candidate antecedents with Kohonen neural network optimizes the performance of the model during training and results in a better F1 for the proposed model than previous coreference resolution models. Also, using a fuzzy method for weighting features leads to the functioning of coreference resolution with higher accuracy which was a shortcoming of the previous systems. Moreover, using bidirectional GRU for finding long dependencies of words in spans works better than other RNN-based networks. The paper also makes a comparison between the proposed approach and coreference resolution based systems. The proposed approach properly manages the problem of coreference resolution with the lowest error rate. Additionally, the precision and recall values of the previous approaches were lower than those of the proposed approach. Although, the previous approaches showed slightly better performance on some datasets for a particular language, the proposed approach shows better performance for different types of data approaches CoNLL-2012 dataset (+3.1 F1) and Yahoo's news site (+6.6 F1). The F1 in named entity recognition rate on the English Gigaword dataset improved by 7.04. We suggest the following future research ideas:

1. Examination of other word embedding methods such as XLNet [46] as the input of a bi-GRU network.
2. Using other new RNNs for extracting spans from input vectors.
3. Using knowledge resources such as medical, syntactic, semantic, and linguistic knowledge for better word representation.
4. Using other criteria weighting methods such as MACBETH [47], DCE [48], PAPRIKA [49], etc.
5. Using Fuzzy-MCDM methods for ranking candidate antecedents.

CONFLICT OF INTEREST

The authors declare they have no conflicts of interest.

AUTHORS' CONTRIBUTIONS

Samira Hourali contributed to state of the art and model design, implementation, results analysis, writing - review & editing. Morteza Zahedi contributed to review & editing. Mansour Fateh contributed to review & editing.

Funding Statement

This research received no external funding.

REFERENCES

[1] K. Lee, L. He, L. Zettlemoyer, Higher-Order coreference resolution with coarse-to-fine inference in Proceeding of 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, Louisiana, 2018, vol. 2, pp. 687–692.

[2] M. Joshi, O. Levy, D.S. Weld, L. Zettlemoyer, BERT for coreference resolution: baselines and analysis in Proceeding of 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 2019, pp. 5802–5807.

[3] B. Kantor, A. Globerson, Coreference resolution with entity equalization, in Proceeding of 57th Conference of the Association for Computational Linguistics, Florence, Italy, 2019, pp. 673–677.

[4] K. Lee, L. He, M. Lewis, L. Zettlemoyer, End-to-end neural coreference resolution, in Proceeding of 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 2017, pp. 188–197.

[5] K. Clark, C.D. Manning, Deep reinforcement learning for mention-ranking coreference models, in Proceeding of 2016 Conference on Empirical Methods in Natural Language Processing Texas, 2016, pp. 2256–2262.

[6] R. Zhang, C.N. dos Santos, M. Yasunaga, B. Xiang, D. Radev, Neural coreference resolution with deep biaffine attention by joint mention detection and mention clustering, in Proceeding of 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 2018, vol. 2, pp. 102–107.

[7] H. Fei, X. Li, D. Li, P. Li, End-to-end deep reinforcement learning based coreference resolution, in Proceeding of 57th Conference of the Association for Computational Linguistics, Florence, Italy, 2019, pp. 660–665.

[8] Y. Liu, *et al.*, RoBERTa: a robustly optimized BERT pretraining approach, arXiv: 1907.11692, 2019.

[9] H.A. Simon, The New Science of Management Decision, Harper & Row, New York, 1960.

[10] S. Pradhan, A. Moschitti, N. Xue, O. Uryupina, Y. Zhang, CoNLL-2012 shared task: modeling multilingual unrestricted coreference in OntoNotes, in Joint Conference on EMNLP and CoNLL-Shared Task, Jeju, Republic of Korea, 2012, pp. 1–40.

[11] R. Grishman, B. Sundheim, Message understanding conference-6: a brief history, in COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics, Copenhagen, Denmark, 1996, pp. 466–471.

[12] C. Napoles, M. Gormley, B. Van Durme, Annotated gigaword, in Proceeding of Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction, Montréal, Canada, 2012, pp. 95–100.

[13] M. Sun, Y. Liu, Z. Liu, M. Zhang, Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, Springer, New York, 2015.

[14] V. Ng, C. Cardie, Improving machine learning approaches to coreference resolution, in 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, 2002, pp. 104–111.

[15] K. Clark, C.D. Manning, Entity-centric coreference resolution with model stacking, in Proceeding of 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 2015, vol. 1, pp. 1405–1415.

[16] S. Wiseman, A.M. Rush, S.M. Shieber, Learning global features for coreference resolution, in Proceeding of NAACL-HLT, San Diego, California, 2016, pp. 994–1004.

[17] G. Durrett, D. Klein, Easy victories and uphill battles in coreference resolution, in Proceeding of 2013 Conference on Empirical Methods in Natural Language Processing, Seattle, 2013, pp. 1971–1982.

[18] S.J. Wiseman, A.M. Rush, S.M. Shieber, J. Weston, Learning anaphoricity and antecedent ranking features for coreference resolution, in Proceeding of 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 2015, vol. 1, pp. 1416–1426.

[19] A. Haghighi, D. Klein, Coreference resolution in a modular entity-centered model, in Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Los Angeles, California, 2010, pp. 385–393.

[20] K. Clark, C.D. Manning, Improving coreference resolution by learning entity-level distributed representations, in Proceeding of 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 2016, vol. 1, pp. 643–653.

[21] S.M. Harabagiu, From lexical cohesion to textual coherence: a data driven perspective, Int. J. Pattern. Recognit. 13 (1999), 247–265.

[22] N.S. Moosavi, M. Strube, Lexical features in coreference resolution: to be used with caution, in Proceeding of 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, Canada, 2017, vol. 2, pp. 14–19.

[23] T. Liang, D.-S. Wu, Automatic pronominal anaphora resolution in English texts, Int. J. Comp. Ling. Chin. Lang. Proc. 9 (2004), 21–40.

[24] J.R. Hobbs, Resolving pronoun references, Lingua. 44 (1978), 311–338.

[25] R. Kibble, A reformulation of rule 2 of centering theory, Comput. Linguist. 27 (2001), 579–587.

[26] A. Zeldes, S. Zhang, When annotation schemes change rules help: a configurable approach to coreference resolution beyond OntoNotes, in Proceeding Workshop on Coreference Resolution Beyond OntoNotes (CORBON 2016), San Diego, 2016, pp. 92–101.

[27] H. Lee, A. Chang, Y. Peirsman, N. Chambers, M. Surdeanu, D. Jurafsky, Deterministic coreference resolution based on entity-centric, precision-ranked rules, Comput. Linguist. 39 (2013), 885–916.

[28] A. Haghighi, D. Klein, Simple coreference resolution with rich syntactic and semantic features, in Proceeding of 2009 Conference on Empirical Methods in Natural Language Processing, Singapore, 2009, pp. 1152–1161.

[29] J.R. Finkel, C.D. Manning, Enforcing transitivity in coreference resolution, in Proceeding of 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers, Columbus, 2008, pp. 45–48.

[30] P. Denis, J. Baldridge, Joint determination of anaphoricity and coreference resolution using integer programming, in Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, Rochester, 2007, pp. 236–243.

[31] A. Haghighi, D. Klein, Unsupervised coreference resolution in a nonparametric Bayesian model, in Proceeding of 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, 2007, pp. 848–855.

[32] V. Ng, Unsupervised models for coreference resolution, in Proceeding of Conference on Empirical Methods in Natural Language Processing, Honolulu, 2008, pp. 640–649.

[33] X. Peng, R. Krishankumar, K.S. Ravichandran, Generalized orthopair fuzzy weighted distance-based approximation (WDBA) algorithm in emergency decision-making, Int. J. Intell. Syst. 34 (2019), 2364–2402.

[34] X. Peng, J. Dai, H. Garg, Exponential operation and aggregation operator for q-rung orthopair fuzzy set and their decision-making method with a new score function, Int. J. Intell. Syst. 33 (2018), 2255–2282.

[35] H.T. Nguyen, T.H. Cao, Named entity disambiguation: a hybrid approach, Int. J. Intell. Syst. 5 (2012), 1052–1067.

[36] J. Gehring, M. Auli, D. Grangier, D. Yarats, Y.N. Dauphin, Convolutional sequence to sequence learning, in Proceeding of 34th International Conference on Machine Learning, Sydney, Australia, 2017, pp. 1243–1252.

[37] P.P. Bonissone, K.S. Decker, Selecting uncertainty calculi and granularity: an experiment in trading-off precision and complexity, Mach. Intell. Patt. Rec. 4 (1986), 217–247.

[38] P.P. Bonissone, A fuzzy sets based linguistic approach: theory and applications, in Proceeding of 12th Conference on Winter simulation, Orlando, 1980, pp. 99–111.

[39] T. Kohonen, Self-Organization and Associative Memory, Springer Science & Business Media, Berlin, 2012.

[40] M. Vilain, J. Burger, J. Aberdeen, D. Connolly, L. Hirschman, A model-theoretic coreference scoring scheme, in Proceeding of 6th Conference on Message Understanding, Columbia, 1995, pp. 45–52.

[41] X. Luo, On coreference resolution performance metrics, in Proceeding of Conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia, Canada, 2005, pp. 25–32.

[42] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall PTR, Upper Saddle River, New Jersey, 1994.

[43] J. Pennington, R. Socher, C. Manning, Glove: global vectors for word representation, in Proceeding of 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 2014, pp. 1532–1543.

[44] M. Peters, *et al.*, Deep contextualized word representations, in Proceeding of 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, 2018, vol. 1, pp. 2227–2237.

[45] K.-W. Chang, R. Samdani, A. Rozovskaya, M. Sammons, D. Roth, Illinois-Coref: the UI system in the CoNLL-2012 shared task, in Joint Conference on EMNLP and CoNLL-Shared Task, Jeju Island, Korea, 2012, pp. 113–117.

[46] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, Q.V. Le, XLNet: generalized autoregressive pretraining for language understanding, arXiv: 1906.08237, 2019.

[47] C.A. Bana E Costa, J.-C. Vansnick, Applications of the MACBETH approach in the framework of an additive aggregation model, JMCDA. 6 (1997), 107–114.

[48] K. Marsh, P. Dolan, J. Kempster, M. Lugon, Prioritizing investments in public health: a multi-criteria decision analysis, J. Public Health. 35 (2012), 460–466.

[49] P. Hansen, F. Ombler, A new method for scoring additive multi-attribute value models using pairwise rankings of alternatives, JMCDA. 15 (2008), 87–107.