Ensemble ranking

Aggregation of rankings produced by different multi-criteria decision-making methods

Mohammadi, Majid; Rezaei, Jafar

DOI
10.1016/j.omega.2020.102254

Publication date
2020

Document Version
Final published version

Published in
Omega (United Kingdom)

Omega 96 (2020) 102254

Contents lists available at ScienceDirect

Omega

journal homepage: www.elsevier.com/locate/omega

Ensemble ranking: Aggregation of rankings produced by different multi-criteria decision-making methods[☆]

Majid Mohammadi[a,b,*], Jafar Rezaei[a]

[a] Faculty of Technology, Policy, and Management, Delft University of Technology, The Netherlands
[b] The Jheronimus Academy of Data Science, s-Hertogenbosch, The Netherlands

ARTICLE INFO

Article history:
Received 23 July 2019
Accepted 20 March 2020
Available online 25 March 2020

Keywords:
MCDM
Half-quadratic
Ensemble ranking
Ontology alignment

ABSTRACT

One of the essential problems in multi-criteria decision-making (MCDM) is ranking a set of alternatives based on a set of criteria. In this regard, there exist several MCDM methods which rank the alternatives in different ways. As such, it would be worthwhile to try and arrive at a consensus on this important subject. In this paper, a new approach is proposed based on the half-quadratic (HQ) theory. The proposed approach determines an optimal weight for each of the MCDM ranking methods, which are used to compute the aggregated final ranking. The weight of each ranking method is obtained via a minimizer function that is inspired by the HQ theory, which automatically fulfills the basic constraints of weights in MCDM. The proposed framework also provides a consensus index and a trust level for the aggregated ranking. To illustrate the proposed approach, the evaluation and comparison of ontology alignment systems are modeled as an MCDM problem and the proposed framework is applied to the ontology alignment evaluation initiative (OAEI) 2018, for which the ranking of participating systems is of the utmost importance.

1. Introduction

Multi-criteria decision-making (MCDM) is a branch of Operations Research that has numerous applications in a variety of areas involving real decision-making problems. In a typical MCDM problem, K alternatives are evaluated on the basis of n criteria, and the outcome of the evaluation is summarized in a so-called performance matrix, within which MCDM methods are used to select the best, sort, or rank the alternative(s). The focus of this study is on ranking, where a set of K alternatives needs to be ranked. There exist several MCDM methods which can be used for the ranking problem, including value and utility-based methods such as AHP (analytic hierarchy process) [48], ANP (analytic network process) [49], BWM (best-worst method) [47], SMART (simple multi-attribute rating technique) [14], and Swing [36], and also the outranking methods like ELECTRE (ELimination and Choice Expressing REality) and its extensions [17], and PROMETHEE (Preference Ranking Organization METHod for Enrichment of Evaluations) and its extensions [7]. For more information about popular MCDM methods, see [55].

One of the main controversial issues in this area is that different MCDM methods, even when they use the same input, produce different and potentially conflicting rankings, which means that finding an overall aggregated ranking of alternatives is of the essence. Some studies ignore the existence of such a conflict [29], or use a simple ranking statistic, like averages [43], while yet other methods attempt to reconcile the difference and work out a compromise [28,42]. Ku et al. [28] estimate the weight for each MCDM method based on the Spearman's correlation coefficient. The underlying idea is that if the ranking of an MCDM method deviates from those of other methods, it would then be assigned a lower weight. As such, the weight of each MCDM ranking is computed using the correlation coefficient. By the same token, Ping et al. [42] has proposed an optimization problem to determine the weight of each individual MCDM method and then aggregate them accordingly. The optimization problem assumes that the final aggregated ranking is a weighted linear combination of the rankings provided by different MCDM methods, and it tries to determine the weights accordingly. Although these methods do come up with a final aggregated ranking, they do not provide any further information about the consensus or reliability of the aggregated ranking.

In this paper, a new ensemble method is proposed based on the half-quadratic (HQ) theory [18,19,37]. In this regard, a new model is proposed based on a general non-convex HQ function,

[☆] This manuscript was processed by Associate Editor Triantaphyllou.

[*] Corresponding author.

E-mail address: majid.mohammadi690@gmail.com (M. Mohammadi).

https://doi.org/10.1016/j.omega.2020.102254

and the procedure involved in determining the optimal solution to the given minimization is provided with guaranteed convergence. Although no weights for the MCDM methods are considered explicitly, the proposed model estimates a weight for each of the MCDM methods by using the so-called *minimizer* function inspired by the HQ theory, whose estimation improves adaptively throughout the optimization procedure. An MCDM method whose ranking is different from those of most of the other MCDM methods being used is treated as an outlier in the proposed framework and, as such, is assigned a lower weight. The aggregated final ranking is also obtained by the weighted combination of rankings of the MCDM methods being used, which means that the methods whose rankings deviate from others will have a lower impact on the final ranking. Although the proposed model is unconstrained, interestingly, the computed weights by the minimizer function preserve the non-negativity and unit-sum properties, that are required for the MCDM methods. The proposed compromise method is also objective, since it does not need to elicit preferences from decision-makers. However, the MCDM methods being used in the framework could belong to either class of MCDM methods (subjective or objective).

For some of the HQ functions, there are parameters that have to be tuned. To that end, we take advantage of several recent studies to tune the parameters efficiently [22,24]. Having such parameters helps compute a consensus index and trust level based on the computed weights. The outcome of the proposed method is to determine the weights of MCDM methods and compute the final aggregated ranking of alternatives, as well as two indicators showing the level of agreement and reliability of the final aggregated ranking.

As a real-world implementation, we study the evaluation and comparison of ontology alignment systems by using different MCDM methods. Such a comparison is of the essence for two major reasons. First, there are numerous ontology alignment systems in the existing literature [13,16,25,35,46,59], each claiming to be superior to the other available systems. To support that claim, the developers of the systems involved typically look at solely one performance score, on which the claim of superiority is based. If there are multiple benchmarks, the average of these scores is computed and regarded as the overall performance representation. However, the main drawback of using averages is that it only allows a comparison on the basis of one performance score. As a result, it is not possible to take into account different facets of a system measured by several metrics. For instance, an important criterion for alignment is execution time, which also has to be included in an evaluation and comparison. Here, we formulate the comparison of ontology alignment systems as an MCDM problem, where the performance metrics are the criteria, and the ontology alignment systems are the alternatives. Consequently, the decision which system is superior is transformed into an MCDM problem, making it possible to compare the systems based on multiple metrics. The second reason for using MCDM methods to assess alignment systems is the competition that exists in the ontology alignment evaluation initiative (OAEI), with several standard benchmarks in divided tracks with an available reference (or gold standard). Within that competition, the participating systems conduct the alignment on the given ontologies, and their outcome is then juxtaposed with the reference for evaluation. In addition, there are various performance metrics for different benchmarks, making the final ranking of the systems, which is potentially one of the principal goals of the competition in the first place, much more difficult. In this paper, we review the performance metrics for five OAEI tracks, and apply the MCDM methods along with the proposed ensemble method to determine the final ranking of the systems. The methodology proposed in this paper can also be used by the OAEI organizers to evaluate the participating systems with respect to multiple performance metrics.

In summary, this paper makes the following contributions:

- A new approach for ensemble ranking is proposed based on the HQ theory.
- The proposed method can assign weights objectively to the MCDM methods being used, since no decision-maker is involved in determining the weights of the final aggregated ranking.
- The proposed method can also be used to compute a consensus index and a trust level for the final aggregated ranking.
- As a real-world implementation, we study the ranking of ontology alignment systems with respect to multiple performance metrics. Such a ranking is of the utmost importance, particularly for the OAEI where there is a competition involving several standard benchmarks. The proposed ensemble method can be used in other ontology alignment benchmarks as well as any other MCDM problem that uses multiple MCDM methods.

The remainder of this article is structured as follows. In Section 2, we present the proposed ensemble method, followed by an overview of MCDM methods being used in Section 3. Sections 4 and 5 are devoted to our real-world implementation of the proposed method in ontology alignment, while the lessons learned are discussed in Section 6, and conclusions and future research directions are presented in Section 7. The MATLAB code and the MS Excel solver of the proposed method are freely available at https://github.com/Majeed7/EnsembleRanking.

2. Ensemble ranking: A half-quadratic programming approach

The MCDM methods may provide different rankings for the same problem because they use different mechanisms, making it hard to provide sufficient support for the ranking of one MCDM method compared to the others. As such, in this section, a compromise method is developed to estimate the final ranking of all alternatives based on the rankings of different MCDM methods. The proposed method utilizes the HQ theory which results in estimating a weight for each of the MCDM methods. The weights obtained by the method satisfy the non-negativity and unit-sum properties, which are necessary for the MCDM methods. In addition, the proposed method is objective, since the weights are computed without any expert input. Another important property of the proposed method is that, in contrast to averaging, it is insensitive to outliers, owing to the use of the robust HQ functions. For aggregating MCDM rankings, outliers are indeed the rankings that are different from the majority of rankings, which means that it is to be expected that they contribute less to the final aggregated ranking. In addition to the aggregated ranking, a consensus index and a trust level are calculated for the aggregated ranking. In the following, we first explain the notations used in the study which follows by reviewing the fundamentals of the HQ theory.

We begin by explaining the notations used in this article. The alternatives are referred to as A_i, $i = 1, 2, \ldots, K$, while the performance metrics or criteria are denoted by P_j, $j = 1, 2, \ldots, n$. Thus, there are K alternatives which are evaluated with respect to n criteria (or performance metrics). Furthermore, the matrix containing all performance scores are shown as X, and $X_{i.}$, $X_{.j}$, X_{ij} referring to the i^{th} row, the j^{th} column, and the element at the i^{th} row and the j^{th} column, respectively. By the same token, the i^{th} element in a vector like s is shown by s_i. Also, we show the Euclidean norm with $\|e\|_2 = \sqrt{\sum_{i=1}^{s} e_i^2}, \forall e \in \mathbb{R}^s$. The ranking of the alternatives computed by the m^{th} MCDM method is shown as R^m, $m = 1, \ldots, M$, and the final aggregated ranking is shown by R^*. In addition, the ranking of alternative k obtained by method m and by the aggregated ranking are shown by R_k^m and R_k^*, respectively.

Table 1
Different M-estimators and their corresponding minimizer function $\delta(.)$ based on the HQ multiplicative form. β is a positive constant, and σ or γ are the parameters of the HQ functions.

estimators	l_1-l_2	fair	log-cosh	Welsch	Huber
HQ function $g(s_j)$	$\sqrt{\beta + \frac{s_j^2}{\sigma^2}} - 1$	$\frac{\lvert s_j\rvert}{\beta} - \log(1 + \frac{\lvert s_j\rvert}{\beta})$	$log(cosh(\beta s_j))$	$1 - \exp(-\frac{s_j^2}{\sigma^2})$	$\begin{cases} \frac{s_j^2}{2}, & \lvert s_j\rvert \le \gamma \\ \gamma\lvert s_j\rvert - \frac{\gamma^2}{2}, & \lvert s_j\rvert > \gamma \end{cases}$
Minimizer Function $\delta(s_j)$	$\frac{1}{\sqrt{\beta+s_j^2}}$	$\frac{1}{\beta(\beta+\lvert s_j\rvert)}$	$\frac{\beta}{s_j}\tanh(\beta s_j)$	$\exp(-\frac{s_j^2}{\sigma^2})$	$\begin{cases} 1, & \lvert s_j\rvert \le \gamma \\ \frac{\gamma}{\lvert s_j\rvert}, & \lvert s_j\rvert > \gamma \end{cases}$

2.1. Half-Quadratic minimization

In this section, we review the fundamental theory of the HQ minimization, introduce the appropriate HQ functions and look at the minimization procedure of the HQ programming.

The Euclidean norm is arguably the most popular loss function used in various circumstances, while least square fitting is the most popular regression technique that utilizes the Euclidean norm as the loss function. Although it is simple and also yields a closed-form solution, it is highly sensitive to outliers and shows diminished performance in noisy environments. A viable way to solve that sensitivity is to use various robust estimators. In robust statistics, M-estimator is a family of the robust estimators, by which the HQ functions are inspired. Although these functions are not convex, their optimum can be obtained using HQ minimization with guaranteed convergence. Table 1 tabulates the HQ functions g(.) along with their minimizer functions $\delta(.)$ that are used in the optimization procedure.

Consider the following minimization,

$$\min_s \sum_j g(s_j), \tag{1}$$

where g(.) is one of the HQ functions tabulated in Table 1. To solve problem (1), there are two forms of the HQ programming (multiplicative [18] and additive [19]) that can efficiently find a local optimal solution. Both forms have been applied to different areas, including robust estimation [34,57], signal processing [33,38,58], image processing [21,23], and machine learning [22,24]. In this paper, we use the multiplicative form since its optimization procedure can be interpreted meaningfully within MCDM.

Based on the multiplicative form of the HQ programming [18,37], problem (1) can be rewritten as

$$\min_{s,w} \sum_j w_j s_j^2 + \psi(w_j), \tag{2}$$

where $w_j > 0$ is the HQ auxiliary variable, and $\psi(.)$ is the convex conjugate of g(.) defined as [5],

$$\psi\left(w_j\right) = \max_e \, ew_j - g(e). \tag{3}$$

To solve minimization (2), variables w and s must be updated iteratively until convergence is reached. Based on the HQ multiplicative theory [18], the update of variables is as follows:

$$w_j^{l+1} = \delta(s_j^l),$$
$$s^{l+1} = \arg\min_s \sum_j w_j^{l+1} s_j^2, \tag{4}$$

where $\delta(.)$ is the minimizer function with respect to g(.) (see Table 1), and l and $l+1$ represent the iteration counter.

In the next section, a new compromise method is developed based on the multiplicative HQ minimization, and it is shown that the auxiliary variable w would play the role of weights in the MCDM problems. Since the value of w is reliant on the type of HQ function $g(.)$, different HQ functions would result in different weights and different final aggregated ranking. We particularly consider the Welsch M-estimator, for two reasons. First, it has shown a promising performance in a variety of problems and it is known to be the most promising and outlier-robust estimator among the HQ functions [23]. Second, we can calculate a consensus index and a trust level if the Welsch estimator is used.

2.2. An HQ-based compromise method

The proposed ensemble method can be used for any number of MCDM methods. In this regard, assume that there are M MCDM methods which rank K alternatives on the basis of n criteria.

A simple yet practical solution to estimate the overall ranking R^* is to minimize its Euclidean distance to each computed ranking. The corresponding minimization is,

$$\min_{R^*} \frac{1}{2} \sum_{m=1}^{M} \|R^m - R^*\|_2^2, \tag{5}$$

where M is the number of MCDM methods and R^m is the ranking of the m^{th} MCDM method. Minimization (5) has the following closed-form solution,

$$R^* = \frac{1}{M} \sum_{m=1}^{M} R^m, \tag{6}$$

which is indeed the average of the rankings produced by different methods. However, averages are not reliable estimators, since they are sensitive to outliers [11], like other methods using the Euclidean norm as their basic loss function. In aggregating rankings, it means that, if one MCDM method has a distinct ranking from the other methods, it can significantly influence the aggregated ranking. Instead, we utilize the HQ functions, which are potentially insensitive to outliers [26], as well as allowing us to compute a consensus index and trust level for the final aggregated ranking.

The proposed optimization problem to estimate R^* is,

$$\min_{R^*} \frac{1}{2} \sum_{m=1}^{M} g(\|R^m - R^*\|_2), \tag{7}$$

where g(.) is an HQ function. Although minimization (7) is not convex, it can be solved efficiently using half-quadratic programming [18,37]. Using the HQ multiplicative form as in equation (2), minimization (7) can be restated as,

$$\min_{R^*,\alpha} J(R^*, \alpha) = \sum_{m=1}^{M} \alpha_m \|R^m - R^*\|_2^2 + \psi(\alpha_m), \tag{8}$$

where $\alpha \in R^M$ is the half-quadratic auxiliary variable. According to the HQ programming, the following steps must be iterated until convergence for the two variables is reached,

$$\alpha_m = \delta\Big(\|R^m - R^*\|_2\Big), \quad m = 1, \ldots, M,$$

$$R^* = \arg\min_{R^*} \sum_{m=1}^{M} \alpha_m \|R^m - R^*\|_2^2. \tag{9}$$

The solution to the first step is obtained by the minimizer function tabulated in Table 1, and the optimum for the second step is obtained by setting the derivative of the objective function equal to zero, i.e.,

$$\frac{dJ}{dR^*} = 0 \Rightarrow \sum_{m=1}^{M} \alpha_m (R^m - R^*) = 0$$
$$\Rightarrow R^* \sum_{m=1}^{M} \alpha_m = \sum_{m=1}^{M} \alpha_m R^m$$
$$\Rightarrow R^* = \sum_{m=1}^{M} w_m R^m, \quad \text{where } w_m = \frac{\alpha_m}{\sum_{j=1}^{M} \alpha_j}. \tag{10}$$

Thus, the final aggregated ranking is computed as the weighted sum of all the MCDM rankings, with the weights being computed by the minimizer function. Interestingly, the weights of MCDM rankings in (10) are non-zero and fulfill the unit-sum property, which are the requirements for the MCDM methods. Note that the optimization problem is unconstrained and these properties are fulfilled, thanks to the use of the HQ functions.

Algorithm 1 summarizes the overall procedure of the proposed ensemble ranking of MCDM methods.

Algorithm 1 Ensemble Ranking.

```
**Input:** Rankings $R^m$, $m = 1, 2, \dots, M$.
**while** *NotCongverged* **do**
    $\alpha_m = \delta(\|R^m - R^*\|_2), \quad m = 1, 2, \dots, M$
    $w_m = \alpha_m / \sum_j \alpha_j, \quad m = 1, 2, \dots, M$
    $R^* = \sum_m w_m R^m$
**end while**
**Output** Final Ranking $R^*$, $\alpha$
```

The following lemma guarantees the convergence of this algorithm.

Lemma 2.1. *The sequence* $\{(\alpha^l, R^{*l}), l = 1, 2, \dots\}$ *generated by Algorithm 1, where l indicates the iteration number, converges.*

Proof. The function $\delta(.)$ has the following property [37],

$$J(\alpha^{l+1}, R^{*l+1}) \le J(\alpha^l, R^{*l+1}), \tag{11}$$

where R^* is assumed to be fixed. Similarly, the sequence of R^* is decreasing since J is convex, e.g.,

$$J(\alpha^{l+1}, R^{*l+1}) \le J(\alpha^{l+1}, R^{*l}). \tag{12}$$

Thus, the sequence

$$\{\dots, J(\alpha^l, R^{*l}), J(\alpha^{l+1}, R^{*l}), J(\alpha^{l+1}, R^{*l+1}), \dots\}$$

converges as $l \to \infty$ since J is bounded. □

Remark 2.2. The proposed ensemble method is predicated on the fact that proper ranking methods are used, since the final aggregated ranking is naturally dependent on the ranking methods in question. If we add or remove a ranking method, the aggregated ranking is likely to change. However, in cases which include a significant number of methods, the proposed method is much less sensitive to adding or removing a ranking method. As such, the proposed method can be particularly useful in voting systems which usually contain a considerable number of votes.

Remark 2.3. The methods for ensemble ranking are useful for the case where there is no prior information about the suitability of one specific ranking method. In this situation, the rankings of different methods are treated equally a priori, and finding an aggregated ranking is desired, typically by working out a compromise between different rankings.

2.3. Consensus index and trust level

The weight of each MCDM method differs with respect to the HQ function in question, since $\delta(.)$ relies on the $g(.)$ function. Consequently, various HQ functions would result in different weights and a different final aggregated ranking. Among the HQ functions, the Welsch estimator has shown a promising performance in a number of domains [22,24]. Interestingly, it is possible to obtain a consensus index and trust level using this estimator, owing to its use of the Gaussian distribution in the formulation. Prior to obtaining the consensus index and trust level, we first need to discuss tuning the parameter σ in the Welsch estimator. As a recent study has indicated [24], the parameter of this estimator can be tuned recursively in each iteration as,

$$\sigma = \frac{\sum_{m=1}^{M} \|R^m - R^*\|_2^2}{2K^2}. \tag{13}$$

After computing σ in the optimization procedure, we now discuss the consensus index and the trust level of the final ranking obtained by Algorithm 1.

Definition 2.4 (Consensus Index)**.** A consensus index C shows the extent to which all MCDM methods agree upon the final ranking.

The key element in this definition is that the consensus index shows the agreement among all the ranking methods being used, allowing us to compute the similarity of each ranking with the final aggregated ranking, thanks to the Welsch estimator. As a result, the consensus index C of a given final ranking R^* with respect to rankings R^m, $m = 1, 2, \dots, M$ can be computed as,

$$C(R^*) = \frac{1}{KM} \sum_{k=1}^{K} \sum_{m=1}^{M} q_{km}, \quad q_{km} = \frac{\mathcal{N}_\sigma(R_k^* - R_k^m)}{\mathcal{N}_\sigma(0)}, \tag{14}$$

where $\mathcal{N}_\sigma(.)$ is the probability density function of the Gaussian distribution with a mean of zero and a standard deviation of σ, and $\mathcal{N}_\sigma(0)$ is used to normalize the similarity computation, thus $q_{km}, C(R^*) \in [0, 1]$. If there is a complete agreement between different rankings, then

$$q_{km} = \frac{\mathcal{N}_\sigma(0)}{\mathcal{N}_\sigma(0)} = 1, \quad \forall k, m, \sigma,$$

that results in a consensus index of one. As rankings deviate from each other, the consensus index decreases. As a result, the consensus index is an indicator of the agreement among different rankings. It means that, if there is one ranking method that is different from the rest, it can adversely affect the consensus index. At the same time, this distinct ranking method is treated as an outlier in the HQ functions being used. As a result, it will have less impact on the final ranking, while it can profoundly influence the consensus index.

Definition 2.5 (Trust Level)**.** A trust level T for ensemble ranking is the degree to which one can accredit the final aggregated ranking.

The trust level is an indicator of reliability of the final ranking. For instance, if there is an MCDM ranking that deviates significantly from the majority of rankings, it takes a lower weight in Algorithm 1, and consequently, has less of an impact on the final ranking. Since the weight of such a method is lower than that of the other methods, it should also have less impact on the trust level. Taking this into account, the trust level can be computed as,

$$T(R^*) = \frac{1}{K} \sum_{k=1}^{K} \sum_{m=1}^{M} w_m q_{km}, \tag{15}$$

where $w_m, m = 1, \dots, M,$ is computed in Algorithm 1. Thus, the trust level is distorted to a lesser extent by the rankings that

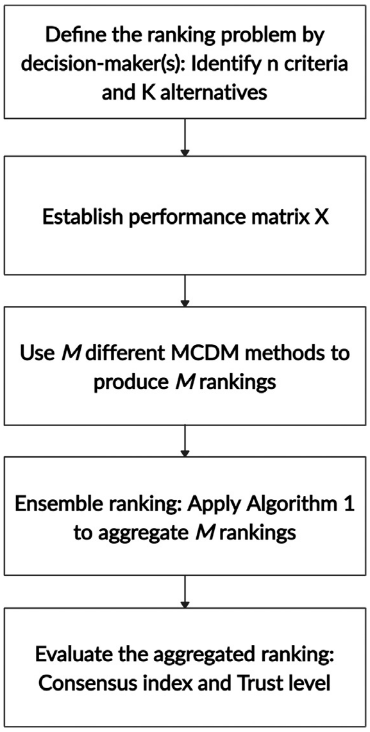
Define the ranking problem by decision-maker(s): Identify n criteria and K alternatives
Establish performance matrix X
Use *M* different MCDM methods to produce *M* rankings
Ensemble ranking: Apply Algorithm 1 to aggregate *M* rankings
Evaluate the aggregated ranking: Consensus index and Trust level

Fig. 1. The implementation process of the proposed ensemble ranking to a decision-making problem.

are different from the majority of rankings, and it is a measurement of the reliability of the aggregated ranking R^* computed by Algorithm 1. It is evident from equation (15) that the trust level is equivalent to the consensus index if the weights of MCDM methods, i.e., $w_m, m = 1, 2, \dots, M,$ are identical.

Fig. 1 summarizes the implementation process of the proposed ensemble ranking to a decision-making problem.

3. Three MCDM methods for illustrating the proposed approach

There exist several MCDM methods which can be used for the ranking problem (see [55] for an overview). In this study, three different MCDM methods (TOPSIS, VIKOR, and PROMETHEE) are selected to illustrate the proposed ensemble ranking method. These methods are used (in the next section) to rank alignment systems with respect to several performance metrics (criteria). We selected these three methods as they are among popular methods in the MCDM field (see, for instance, [12,32,44] for the applications of TOPSIS, [2,4,50] for the applications of VIKOR, and [3,20,31] for the applications of PROMETHEE). Secondly, compared to many other MCDM methods, they can be used in an objective way, without having to include the opinions of experts or users. In addition, they were selected because of their ability to rank alternatives, which implies that other MCDM methods, which are devised for other purposes (such as sorting or selecting), are not appropriate for this study, although that does not mean that the three MCDM methods being used in this study are the only usable methods, nor does the proposed method rely on the number of MCDM methods.

3.1. Technique for order preference by similarity to ideal solution (TOPSIS)

TOPSIS is one of the popular MCDM methods for ranking alternatives with respect to a set of criteria [56]. It first identifies the positive-ideal and negative-ideal solutions and then ranks the alternatives based on their distances to the two computed solutions. The alternatives are ranked based on their closeness to the positive-ideal solution and their distance from the negative-ideal solution.

While TOPSIS has many variations and extensions [1,8,10], in this study, we adopt the original version proposed in [41]. The ranking process in TOPSIS includes the following steps:

Step 1: First, the performance matrix should be normalized. The elements of the normalized matrix $\hat{X}$ are calculated as,

$$\hat{X}_{kj} = \frac{X_{kj}}{\|X_{.j}\|}, \quad k = 1, 2, \dots, K, \quad j = 1, 2, \dots, n. \tag{16}$$

Step 2: Find the positive-ideal solution $S^+ = (S_1^+, S_2^+, \dots, S_n^+)$, where $S_j^+ = \max_k \hat{X}_{kj}$ for benefit criteria, e.g., profit, and $S_j^+ = \min_k \hat{X}_{kj}$ for cost criteria, e.g., time.

Step 3: Find the negative-ideal solution $S^- = (S_1^-, S_2^-, \dots, S_n^-)$, where $S_j^- = \min_k \hat{X}_{kj}$ for benefit criteria, and $S_j^- = \max_k \hat{X}_{kj}$ for cost criteria.

Step 4: Calculate the Euclidean distance to the positive-ideal and negative-ideal solutions for each alternative. For the k^{th} alternative, the distance to the ideal solution, D_i^+, and to the negative-ideal solution, D_i^-, is computed as

$$D_k^+ = \|\hat{X}_{k.} - S^+\|, \qquad D_k^- = \|\hat{X}_{k.} - S^-\|. \tag{17}$$

Step 5: Calculate the ratio L_k for each alternative as

$$L_k = \frac{D_k^-}{D_k^+ + D_k^-}, \quad k = 1, \dots, K. \tag{18}$$

Step 6: Rank the alternatives according to their ratios L_k in a descending order.

3.2. Vlsekriterijumska optimizacija i kompromisno resenje (VIKOR)

VIKOR is another MCDM method that ranks the alternatives based on a set of possibly conflicting criteria. The procedure used in VIKOR can be summarized as follows [39,40].

Step 1: Find the best f^+ and the worst f^- values among the alternatives for all criteria. For the benefit criteria, we have

$$f_j^+ = \max_i X_{ij}, \quad j = 1, 2, \dots, n,$$
$$f_j^- = \min_i X_{ij}, \quad j = 1, 2, \dots, n, \tag{19}$$

where the minimum and maximum are substituted if it is the cost criteria.

Step 2: For each alternative, compute S_i and R_i as

$$S_i = \sum_{j=1}^{n} \frac{f_j^+ - X_{ij}}{f_j^+ - f_j^-},$$
$$R_i = \max_j \left\{ \frac{f_j^+ - X_{ij}}{f_j^+ - f_j^-} \right\}. \tag{20}$$

Step 3: For each alternative, calculate Q_i as

$$Q_i = \nu \frac{S_i - S^+}{S^- - S^+} + (1 - \nu) \frac{R_i - R^+}{R^- - R^+},$$
$$S^+ = \min_i S_i, \qquad S^- = \max_i S_i,$$
$$R^+ = \min_i R_i, \qquad R^- = \max_i R_i, \tag{21}$$

where $\nu \in [0, 1]$ is a trade-off parameter. It is the common practice to set $\nu = 0.5$.

Step 4: Ranking the alternatives based on their corresponding Q_i in descending order.

Step 5: For two alternatives A_i and A_k, A_i is given a better ranking than A_k if: (a) $Q_i - Q_k > 1/(j-1)$; and (b) A_i has a better ranking according to S_i and/or R_i.

3.3. Preference ranking organization METHod for enrichment of evaluations (PROMETHEE)

PROMETHEE uses pairwise comparison between different alternatives to establish a ranking. And while PROMETHEE I [6] conducts partial pairwise comparison and computes the ranking accordingly, PROMETHEE II [54], on the other hand, uses complete pairwise comparison, which is required for the proposed ensemble method and makes it also more suitable to rank the alignment systems. The ranking procedure used by PROMETHEE II is as follows.

Step 1: For $i, k = 1, 2, \ldots, K$, compute the function π_{ik} as the number of criteria in which A_i has better performance than A_k, e.g.,

$$\pi_{ik} = \sum_{j=1}^{n} I(X_{ij} > X_{kj}), \qquad i, k = 1, 2, \ldots, K, \tag{22}$$

where I is the Dirac function which is 1 when the condition in the parenthesis is satisfied, and 0 when it is not.

Step 2: Calculate the positive ϕ^+ and negative ϕ^- outranking flow and the net flow ϕ for each alternative as,

$$\phi^+(A_i) = \frac{1}{K-1}\sum_{k=1}^{K}\pi_{ik}, \qquad \phi^-(A_i) = \frac{1}{K-1}\sum_{k=1}^{K}\pi_{ki}, \tag{23}$$

$$\phi(A_i) = \phi^+(A_i) - \phi^-(A_i). \tag{24}$$

Step 3: Rank in decreasing order the alternatives based on their net flow.

4. Fundamentals of ontology alignment evaluation

In this section, we first review the basic concepts of ontology and ontology alignment, and then discuss the metrics to evaluate the alignment systems.

4.1. Ontology and ontology alignment

An ontology contains the concepts of a domain, along with their properties and relationships. The following definition explains the ontology in a formal manner.

Definition 4.1 (Ontology [15]). An ontology O is a set of the following 4-tuples

$O = (C, Prop, ObjProp, Ins)$

where

- C contains all classes in the ontology representing the concepts;
- *Prop* is the collection of data properties describing the classes within the ontology;
- *ObjProp* is the group of object properties representing the relations of classes within the ontology;
- *Ins* is the set of individuals instantiated from classes, properties, or object properties.

All the classes, properties, and object properties are called the entities of an ontology. The design of an ontology is subjective, so two ontologies describing the same domain can have a distinct structure/terminology, which means that ontology alignment is required to deal with this discrepancy. We now consider the rudimentary concepts of ontology alignment.

Definition 4.2 (Correspondence [15]). To match the ontologies O and O', a correspondence is as a set of 4-tuples

$< e, e', rel, d >$

where

- e and e' are two entities from O and O', respectively;
- *rel* denotes the relation of two entities e and e', e.g., equivalence, subsumption;
- $d \in [0, 1]$ is the degree of the correspondence confidence.

Definition 4.3 (Alignment [15]). Given two ontologies O and O', an alignment is a set of correspondences mapping the concepts of two ontologies in question.

4.2. Performance metrics

Alignment is the typical outcome of the ontology alignment systems, based on which different systems are evaluated and compared. In addition, several standard benchmarks with a known reference alignment have to be included, so that the evaluation can be made by the juxtaposition of the reference and the alignment generated by a system. The three widely-used performance metrics for ontology alignment are precision, recall, and F-measure. Given an alignment A and the reference A^*, precision is the ratio of true positives to the total correspondences in the generated alignment by a system; thus, it can be written as

$$Pr(A, A^*) = \frac{|A \cap A^*|}{|A|} \tag{25}$$

where Pr is the precision and $|.|$ is the cardinality operator.

Recall is another popular metric, which is computed as the ratio of the true positives to the total number of correspondences in the reference. Thus, it can be computed as

$$Re(A, A^*) = \frac{|A \cap A^*|}{|A^*|} \tag{26}$$

where Re is recall.

Both precision and recall represent only one aspect of the alignment systems; the former only considers the correctness of the alignment, while the latter accentuates the completeness of an alignment with respect to the reference. As a combination of both, F-measure is often used. It is the harmonic mean of the precision and recall and is computed as

$$\text{F-measure}(A, A^*) = 2\frac{Pr(A, A^*) \times Re(A, A^*)}{Pr(A, A^*) + Re(A, A^*)}.$$

We do not include F-measure in this study since it is the average of precision and recall, which violates the independence of criteria required for the MCDM methods. Aside from these popular performance metrics, there are two important principles for a given alignment. The first is *conservativity* [52,53], which states that, with regard to the alignment being generated, the system must not impose any new semantic relationship between the concepts of the ontologies involved. The second is *consistency*, which states that the discovered correspondences should not lead to unsatisfiable classes in the merged ontology [53].

There is also a metric called *Recall+*, which indicates the portion of correspondences that a system cannot readily detect. When this performance metric has a higher value, that indicates that the associated system is able to identify the most non-trivial, i.e., non-syntactically identical, correspondences between two given ontologies. In addition, the execution time is another important indicator of the performance of the alignment systems, that also has to be taken into account.

4.3. Participating systems and standard benchmarks: Five OAEI tracks

To determine some of the performance metrics, we need to have the underlying true alignment of the ontologies in question, for which we use the benchmarks of five different tracks of the OAEI whose reference alignment are also available. The tracks

Table 2
The selected performance metrics of five tracks of the OAEI.

OAEI track	Performance metrics/indicators
Anatomy	time, precision, recall, recall+, consistency
Conference	precision, recall, conservativity, consistency
LargeBioMed	time, precision, recall
Disease and Phenotype	time, precision, recall
SPIMBENCH	time, precision, recall

are *anatomy, conference, largeBioMed* (large biomedical track), *disease and phenotype*, and *SPIMBENCH*. By revising the history of the tracks in the OAEI competition[1], as well as asking the organizers of the tracks, the appropriate performance metrics for each of the tracks listed above are obtained. Table 2 tabulates the performance metrics for all five tracks.

According to Table 2, the execution time is essential to all tracks, with the exception of conference, since the size of ontologies in this track is small (i.e., $<$ 100 entities) and the systems are therefore able to perform the alignment swiftly. Furthermore, precision and recall are important in all tracks. However, we did not include F-measure, since it is the harmonic mean of precision and recall. In other words, since the evaluation based on MCDM includes both precision and recall, using F-measure is a redundancy. In addition, the criteria must be independent of each other in MCDM, which means that using F-measure would invalidate the overall ranking computed by various MCDM methods.

The evaluation is conducted on the alignment systems took part in the OAEI 2018. The exhaustive list of the participating systems in one or multiple of the five tracks are AML [16], LogMap, LogMapBio, and LogMapLite [13], SANOM [35], DOME [25], POMAP++ [30], Holontology [45], ALIN [51], XMap [59], ALDO2Vec [46], FCAMapX [9], and KEPLER [27]. Table 3 displays the systems participated in different OAEI tracks. According to this table, 14 systems participated in the anatomy track, 12 in conference, seven in LargeBioMed, eight in disease and phenotype, and three in SPIMBENCH. Another point is that AML and LogMap participated in all five tracks.

5. Experiments

In this section, the MCDM methods and the proposed aggregated methodology are applied to five tracks of the OAEI, and the systems participating in 2018 are compared and ranked accordingly. The alignments produced by various systems are available on the OAEI website.[2]

5.1. Large BioMed Track

The aim of this track is to find alignments between the Foundational Model of Anatomy (FMA), SNOMED CT, and the National Cancer Institute Thesaurus (NCI) ontologies. The ontologies are large and contain tens of thousands of classes. The performance metrics used to rank the systems participated in this track are execution time, precision, and recall.

Table 4 tabulates the ranking of seven systems that applied for matching FMA to NCI. This is an interesting case, since the MCDM rankings are conflicting. In particular, the rankings of VIKOR and PROMETHEE are in line for LogMapBio and FCAMAPX and are both different compared to the ranking of TOPSIS, while the rankings of TOPSIS and VIKOR agree with regard to LogMapLite and XMap and are distinct from the ranking of PROMETHEE. When considering the weights of MCDM methods, it is interesting to see that the weight of VIKOR is relatively high and is close to one, while the weights of the other two methods are lower and close to zero, which means that the proposed ensemble method favors the middle ground ranking among these three MCDM methods. Since two methods have different rankings compared to the aggregated final ranking, the consensus index is not high at around 0.80. At the same time, the trust level is 1.00 because the weights of two MCDM methods are nearly zero so that they cannot affect this indicator. This table shows that AML, LogMap, and XMap are listed as the top three systems in this task.

In addition, Table 5 shows the ranking of participants in matching FMA and SNOMED. This table is similar to Table 4, since VIKOR has a higher weight compared to the other methods, with its ranking situated between the other rankings. The consensus index for the final ranking is 0.80, while the trust level is 0.98. Similarly, Table 6 shows the ranking of seven systems participated in matching NCI to SNOMED. According to this table, VIKOR once more has a higher weight, and as a result, the final consensus index is 0.80, with a trust level of 0.98. According to Tables 5 and 6, AML and LogMap are the top two systems in aligning FMA to SNOMED as well as NCI to SNOMED.

5.2. Disease and Phenotype Track

The OAEI disease and phenotype track comprises matching various disease and phenotype ontologies. The OAEI 2018 consisted of two tasks. The first one to align the human phenotype (HP) ontology to the mammalian phenotype (MP), the second to align the human disease ontology (DOID) and the orphanet and rare diseases ontology (ORDO). The performance metrics used for this track are execution time, precision, and recall.

In the OAEI 2018, eight systems were able to align HP and MP, while nine systems could match DOID and ORDO. Table 7 illustrates the ranking of the systems participated in the OAEI 2018 disease and phenotype track for mapping HP and MP ontologies. According to this table, the weights of TOPSIS and VIKOR are significantly higher than that of PROMETHEE, because the rankings obtained by PROMETHEE deviate more from the other two methods. For instance, PROMETHEE puts AML in the fourth place, while the other two consider it to be the best alignment system. As a result, the weight of PROMETHEE became insignificant. The consensus index for this ranking is 0.85 and its trust level is 0.95. Also, this table indicates that AML, LogMapLite, and LogMap are the top systems in this mapping task.

Another matching task in this track involves the alignment of DOID and ORDO ontologies. Table 8 shows the ranking of the participating systems for this task. According to this table, TOPSIS takes the highest weight, since it is a compromise of the other two MCDM methods. In particular, the TOPSIS ranking of DOME lies between those of VIKOR and PROMETHEE. Also, TOPSIS rankings occasionally agree with one of the other ranking methods: It agrees with VIKOR on ranking LogMap, LogMapLite, and XMap, while it is in line with PROMETHEE with regard to POMAPP++. Given these rankings, TOPSIS has a higher weight compared to other MCDM methods. The consensus index and trust level of this ranking are 0.87 and 0.95, respectively. Accordingly, LogMap, LogMapLite, and XMap are the top systems on this task with regard to all the performance metrics.

5.3. Anatomy track

This track consists of matching the adult mouse anatomy to a part of NCI thesaurus describing the human anatomy. In the OAEI 2018, 14 systems participated in the anatomy track. The systems are compared based on execution time, precision, recall, consistency, and recall+. Table 9 shows the ranking of the systems in the

[1] http://oaei.ontologymatching.org/
[2] http://oaei.ontologymatching.org/2018/results/index.html

Table 3
The OAEI tracks and the participating systems in each individual track for the year 2018.

OAEI track	Alignment systems
Anatomy	LogMapBio, DOME, POMAP++, Holontology, ALIN, AML, XMap, LogMap, ALOD2Vec, FCAMapX, KEPLER, LogMapLite, SANOM, Lily
Conference	Holontology, DOME, ALIN, AML, XMap, LogMap, ALOD2Vec, FCAMapX, KEPLER, LogMapLite, SANOM, Lily
LargeBioMed	AML, LogMap, LogMapBio, XMap, FCAMapX, LogMapLt, DOME
Disease and Phenotype	LogMap, LogMapBio, AML, LogMapLt, POMAP++, Lily, XMap, DOME
SPIMBENCH	AML, Lily, LogMap

Table 4
Ranking of systems taking part in the Large BioMed track for mapping FMA to NCI.

	Time(s)	Precision	Recall	TOPSIS	VIKOR	PROM	R^*	Aggregated ranking
AML	55	0.84	0.87	1	1	1	1	1
LogMap	51	0.86	0.81	2	2	2	2	2
LogMapBio	1072	0.83	0.83	7	6	6	6	6
XMap	65	0.88	0.74	3	3	4	3	3
FCAMapX	881	0.67	0.84	6	7	7	7	7
LogMapLt	6	0.68	0.82	4	4	3	4	4
DOME	12	0.8	0.67	5	5	5	5	5
weights				0.00	1.00	0.00		

* Consensus Index = 0.80
* Trust Level = 1.00

Table 5
Ranking of systems taking part in the Large BioMed track for mapping FMA to SNOMED.

	Time	Precision	Recall	TOPSIS	VIKOR	PROM	R^*	Aggregated ranking
FCAMapX	1736	0.82	0.76	6	5	5	5.00	5
AML	94	0.88	0.69	1	1	1	1.00	1
LogMapBio	1840	0.83	0.65	7	7	6	6.95	7
LogMap	287	0.84	0.64	2	2	4	2.08	2
XMap	299	0.72	0.61	3	6	7	6.02	6
LogMapLt	9	0.85	0.21	5	4	3	3.96	4
DOME	20	0.94	0.20	4	3	2	2.96	3
weights				0.0056	0.9502	0.0442		

* Consensus Index = 0.80
* Trust Level = 0.98

Table 6
Ranking of systems taking part in the Large BioMed track for mapping NCI to SNOMED.

	Time	Precision	Recall	TOPSIS	VIKOR	PROM	R^*	Aggregated ranking
AML	168	0.90	0.67	1	1	1	1	1
FCAMapX	2377	0.80	0.68	6	4	5	4.07	4
LogMapBi	2942	0.85	0.63	7	6	6	6.02	6
LogMap	475	0.87	0.60	3	2	3	2.05	2
LogMapLt	11	0.80	0.57	2	3	4	3.00	3
DOME	24	0.91	0.48	4	5	2	4.90	5
XMap	427	0.64	0.58	5	7	7	6.95	7
weights				0.0255	0.9490	0.0255		

* Consistency Index = 0.80
* Trust Level = 0.98

Table 7
Ranking of eight systems participated in the 2018 OAEI disease and phenotype track. The task involves mapping HP and MP.

	Time	Precision	Recall	TOPSIS	VIKOR	PROM	R^*	Aggregated ranking
LogMap	31	0.88	0.84	2	2	2	2	2
LogMapBio	821	0.86	0.84	3	4	5	3.50	4
AML	70	0.89	0.8	1	1	4	1.01	1
LogMapLt	7	0.99	0.61	4	3	1	3.48	3
POMAP++	1668	0.86	0.58	7	5	7	6.01	6
Lily	4749	0.68	0.65	8	8	8	8	8
XMap	20	0.99	0.31	5	6	3	5.48	5
DOME	46	1	0.31	6	7	6	6.50	7
weights	-	-	-	0.4997	0.4946	0.0057	-	-

* Consensus Index = 0.85
* Trust Level = 0.95

Table 8
Ranking of systems participated in the 2018 OAEI disease and phenotype track. The task involves the alignment of DOID and ORDO.

	Time	Precision	Recall	TOPSIS	VIKOR	PROM	R^*	Aggregated ranking
LogMap	25	0.94	0.78	1	1	4	1.0843	1
LogMapBio	1891	0.9	0.8	6	4	3	5.3494	5
POMAP++	2264	0.87	0.8	7	5	7	6.4337	7
LogMapLt	7	0.99	0.62	2	2	1	1.9718	2
XMap	15	0.97	0.55	3	3	5	3.0562	3
KEPLER	2746	0.88	0.57	8	8	8	8	8
Lily	2847	0.59	0.78	9	9	9	9	9
AML	135	0.51	0.87	5	7	6	5.5943	6
DOME	10	1	0.44	4	6	2	4.5100	4
weights	-	-	-	0.6888	0.2831	0.0281	-	-

* Consensus Index = 0.87
* Trust Level = 0.95

Table 9
Ranking of 14 systems participated in the OAEI 2018 anatomy track.

	Time (s)	Precision	Recall	Recall+	Consist.	TOPSIS	VIKOR	PROM	R^*	Aggregated ranking
LogMapBio	808	0.89	0.91	0.76	1	4	5	4	4.44	4
DOME	22	1	0.62	0.01	0	13	11	7	11.19	11
POMAP++	210	0.92	0.88	0.7	0	6	6	5	5.85	5
Holontology	265	0.98	0.29	0.01	0	14	14	14	14.00	14
ALIN	271	1	0.61	0	1	7	4	11	6.29	6
AML	42	0.95	0.94	0.83	1	1	1	1	1.00	1
XMap	37	0.93	0.87	0.65	1	2	2	2	2.00	2
LogMap	23	0.92	0.85	0.59	1	3	3	3	3.00	3
ALOD2Vec	75	1	0.65	0.09	0	12	10	9	10.66	10
FCAMapX	118	0.94	0.79	0.46	0	8	7	10	7.87	8
KEPLER	244	0.96	0.74	0.32	0	11	12	12	11.60	12
LogMapLite	18	0.96	0.73	0.29	0	9	8	6	8.10	9
SANOM	487	0.89	0.84	0.63	0	5	9	8	7.23	7
Lily	278	0.87	0.8	0.52	0	10	13	13	11.79	13
weights						0.4048	0.4413	0.1539		

* Consensus Index = 0.95
* Trust Level = 0.97

Table 10
Ranking of systems participated in the 2018 OAEI conference track. The evaluation is based on the certain reference alignment.

	Precision	Recall	AvgConserViol	AvgConsisViol	TOPSIS	VIKOR	PROM	R^*	Aggregated ranking
SANOM	0.78	0.76	5.15	4.6	9	4	7	7.67	8
AML	0.83	0.7	1.86	0	3	1	2	2.35	2
LogMap	0.84	0.64	1.19	0	1	2	1	1.04	1
XMap	0.81	0.61	2.65	0.7	4	3	6	5.07	5
KEPLER	0.76	0.61	5.86	7.57	10	9	10	9.96	10
ALIN	0.88	0.54	0.1	0	2	5	3	2.69	3
DOME	0.88	0.54	5.05	0.48	7	7	5	5.88	6
Holontology	0.86	0.55	3.14	0.48	5	6	4	4.49	4
FCAMapX	0.71	0.61	5.9	13	12	12	12	12.00	12
LogMapLite	0.84	0.54	4.57	1.19	6	8	8	7.20	7
ALOD2Vec	0.85	0.54	5.9	1.29	8	10	9	8.65	9
Lily	0.59	0.63	7	6.2	11	11	11	11.00	11
weights					0.3986	0.0436	0.5578		

* Consensus Index = 0.91
* Trust Level = 0.95

anatomy track computed by three MCDM methods, the final ranking being obtained by using the proposed ensemble method. The consensus index and trust level for this track are 0.95 and 0.97, respectively. Based on this table, AML, XMap, and LogMap are the top three systems in the anatomy track.

5.4. Conference Track

The conference track involves matching and aligning seven ontologies from different conferences. For this track, there are two different reference alignments, i.e., certain and uncertain. Table 10 tabulates the result of the analysis of the 12 systems participated in this track at the OAEI 2018 with the certain alignment, with a consensus index of 0.91 and a trust level of 0.95. Based on this table, LogMap, AML, and Alin are the top systems. For the uncertain version of the reference alignment, as Table 11 shows, AML, LogMap, and Holontology are the top three systems. The consensus index and trust level for this track are 0.93 and 0.95, respectively.

5.5. SPIMBENCH Track

The SPIMBENCH task is another matching task, the aim of which is to determine when two OWL instances describe the same Creative Work. There are two datasets, called Sandbox and Mainbox, each of which has a Tbox as the source ontology and Abox as the target. Tbox contains the ontology and instances, and it has to be aligned to Abox, which only contains instances. The difference between Sandbox and Mainbox is that the reference of the

Table 11
Ranking of systems participated in the 2018 OAEI conference track. The evaluation is based on the uncertain reference alignment.

	Precision	Recall	AvgConserViol	AvgConsisViol	TOPSIS	VIKOR	PROM	Average	Aggregated ranking
SANOM	0.8	0.67	5.15	4.6	9	4	4	4.82	5
AML	0.79	0.65	1.86	0	3	1	2	1.67	1
LogMap	0.79	0.58	1.19	0	1	2	3	2.18	2
XMap	0.79	0.55	2.65	0.7	4	3	5	3.85	4
KEPLER	0.68	0.57	5.86	7.57	11	10	9	9.82	10
Holontology	0.81	0.5	0.1	0	2	6	1	3.63	3
ALIN	0.82	0.48	5.05	0.48	7	8	6	7.15	7
FCAMa pX	0.67	0.56	3.14	0.48	5	5	7	5.69	6
DOME	0.82	0.48	5.9	13	12	11	10	10.82	11
ALOD2Vec	0.8	0.49	4.57	1.19	6	7	8	7.18	8
LogMapLite	0.79	0.49	5.9	1.29	8	9	11	9.52	9
Lily	0.58	0.56	7	6.2	10	12	12	11.67	12
weights					0.1639	0.4935	0.3427		

* Consensus Index = 0.93
* Trust Level = 0.95

Table 12
Ranking of systems participated in the 2018 OAEI SPEMBENCH track. The task is Sandbox.

	Precision	Recall	Time	TOPSIS	VIKOR	PROM	R^*	Aggregated ranking
AML	0.83	0.9	6220	2	3	3	3	3
Lily	0.85	1	1960	1	1	1	1	1
LogMap	0.94	0.76	5887	3	2	2	2	2
weights				0	0.50	0.50		

* Consensus Index = 0.77
* Trust Level = 1.00

Table 13
Ranking of systems participated in the 2018 OAEI SPEMBENCH track. The task is Mainbox.

	Precision	Recall	Time	TOPSIS	VIKOR	PROM	R^*	Aggregated ranking
AML	0.84	0.88	37,190	3	3	3	3	3
Lily	0.85	1	3103	1	1	1	1	1
LogMap	0.89	0.71	23,494	2	2	2	2	2
weights				0.33	0.33	0.33		

* Consensus Index = 1.00
* Trust Level = 1.00

former is available to the participants, while the latter is a blind matching task so that participants do not know the real alignment in advance.

There are only three systems included in this track at the OAEI 2018. Tables 12 and 13 list the ranking of the systems for the Sandbox and Mainbox tasks, respectively. The Sandbox task is interesting, since two MCDM methods have identical rankings, while the other, i.e., TOPSIS, differs in ranking two systems, as a result of which its weight becomes insignificant, while the weight of the other two rankings is about 0.50. The consensus index for this ranking is 0.77, while its trust level is 1.00, since the final ranking is identical to the ranking (or average) of the other two MCDM methods.

For the Mainbox task, Table 13 shows the ranking of the three systems on this task. Interestingly, the rankings of the MCDM methods are identical and they all take on a similar weight in the proposed method. As expected, the consensus index and trust level are also one. According to these tables, Lily performs best in both tasks, followed by LogMap and AML.

Remark 5.1. We discussed the ranking of TOPSIS, VIKOR, and PROMETHEE for different OAEI tracks. They all had higher weights in some tracks and lower weights in some of the others. However, the aim of this study is not to compare MCDM methods or discuss their suitability. These methods can take on higher or lower weights in different decision-making problems, and their weights are entirely dependent on the computed rankings based on the performance matrix of the decision-making problem in question.

Remark 5.2. In this study we used three MCDM methods for which we do not need to use the expert/decision-maker opinion to make the final ranking. This, however, does not mean that we cannot use the MCDM methods in which expert/decision-maker opinion is used to make the ranking (such as AHP/ANP, BWM). In fact the rankings (which are the input for our ensemble method) could come from any set of MCDM methods (with or without expert/decision-maker opinion). It is, however, important to know that regardless of the MCDM methods we use in our proposed ensemble method, there is no need to have the opinion of an expert/decision-maker on comparing the rankings which are produced by the different MCDM methods.

6. Discussion

As we discussed earlier, the consensus index and the trust level indicate two different aspects of the final aggregated ranking. Generally speaking, higher values are desirable for both indicators. The consensus index is an indicator of the agreement among all the MCDM methods being used, while the trust level shows the reliability with regard to the final aggregated ranking. Below, based on the main properties of the proposed approach and the findings of the experiments, we elaborate on some general possible outcomes of the proposed methods.

- **Consensus index high, trust level high:** If all the MCDM methods being used have identical rankings, their weights are analogous and equivalent to $1/M$, where M is the number of ranking methods. In this case, the final aggregated ranking is precisely the average of the individual rankings. As a result, the proposed ensemble method represents the average, or equivalently, the HQ functions operate as the Euclidean norm. This is indeed acceptable, since there are no outliers when all the rankings are identical. In this case, because there is full agreement among all the MCDM methods being used, both consensus index and trust level are one.

- **Consensus index low, trust level high:** Where there is a low consensus index and a high trust level, that can mean either of two things. First, if a small fraction of the MCDM methods being used deliver rankings that deviate from the other rankings, the proposed ensemble method treats them as outliers, assigning them lower weights, which reduces their impact on the final aggregated ranking. The presence of such methods can be detected by inspecting the weights obtained by the proposed ensemble method. Methods that have a lower weight are seen as a deviation from the majority of MCDM rankings, as well as from the final ranking, which means they are treated as outliers. The second option is when the number of methods with lower weights is significant compared to the overall number of the MCDM methods being used. The MCDM rankings with higher weights are the intermediates of all the methods. As a result, the intermediate rankings take on higher weights and have a more profound impact on the final aggregated ranking. In both of these cases, the agreement among the MCDM methods being used is low, while the final ranking is fully captured by a fraction of the MCDM methods involved, which is why the consensus index is insignificant and the trust level is high.
- **Consensus index low, trust level low:** If all the MCDM rankings in question deviate significantly from each other, the consensus index will be low. In that case, there is not a share of the MCDM methods involved with significantly higher weights, which means that the trust level is also low.
- **Consensus index high, trust level low:** This scenario does not occur, because the trust level is high when there is a consensus among the MCDM methods being used.

This is a general discussion framework, and we think that the levels could be defined by the decision-makers for a particular problem.

7. Conclusion

In this paper, a new compromise ensemle method was proposed, based on the half-quadratic (HQ) theory. The proposed method can be used to compute a final aggregated ranking, in the form of the weighted sum of the MCDM rankings. The weights in the proposed method were computed using the minimizer functions inspired in the HQ theory, but it satisfied the basic properties of weights in MCDM. In addition, using multiple performance metrics, the ranking of ontology alignment systems was modeled as an MCDM problem, where the systems and the performance metrics served as alternatives and criteria, respectively. In this regard, appropriate MCDM methods were reviewed, each of which could assign a ranking to each system on a benchmark with respect to its performance metrics.

We also introduced two indicators, consensus index and trust level, the former indicates the level of agreement among MCDM ranking methods, while the latter reflects the reliability of the ranking schemes. It became clear in the cases we examined that, when a ranking method deviates from the others, it has a low consensus index but high trust level. As a result, these two indicators are able to delineate different properties of the final aggregated ranking.

Since evaluating and ranking ontology alignment systems are important activities, in particular in light of the ontology alignment evaluation initiative (OAEI) competition, the approach discussed in this article can be used to produce a final ranking of ontology alignment systems in each of the OAEI tracks. The outcome can provide greater insight into the overall performance of systems and promote the report provided annually by the OAEI organizer.

This study can be extended in various ways. To begin with, the performance metrics used to rank the alignment systems are treated as though they are equally important, but it is worthwhile to keep in mind that different performance metrics may in fact not be equally important, which means that one area of future research involves examining the preferences of different performance metrics for different OAEI tracks by the experts in the domain, and then ranking the systems involved accordingly. To that end, a broad range of MCDM methods could be used.

The proposed approach in this paper has the potential to be used for many real-world applications where a number of MCDM methods are used to rank a number of alternatives, and that a consensus among the methods being used are needed to come up with a final aggregated ranking. Finally, we think that it would be interesting to use the proposed method to integrate the votes in voting systems.

CRediT authorship contribution statement

Majid Mohammadi: Conceptualization, Methodology, Software, Writing - original draft. **Jafar Rezaei:** Validation, Writing - review & editing, Supervision.

References

[1] Abo-Sinna MA, Amer AH. Extensions of topsis for multi-objective large-scale nonlinear programming problems. Appl Math Comput 2005;162(1):243–56.
[2] Acuña-Soto CM, Liern V, Pérez-Gladish B. A vikor-based approach for the ranking of mathematical instructional videos. Management Decision 2019;57(2):501–22.
[3] Amaral TM, Costa AP. Improving decision-making and management of hospital resources: an application of the promethee ii method in an emergency department. Oper Res Health Care 2014;3(1):1–6.
[4] Bai C, Rezaei J, Sarkis J. Multicriteria green supplier segmentation. IEEE Trans Eng Manage 2017;64(4):515–28.
[5] Boyd S, Vandenberghe L. Convex optimization. Cambridge university press; 2004.
[6] Brans J. Lingenierie de la decision, llaboration dinstruments daidea la decision. colloque sur laidea la decision. Faculte des Sciences de lAdministration, Universite Laval 1982.
[7] Brans J-P, Mareschal B. Promethee methods. In: Multiple criteria decision analysis: state of the art surveys. Springer; 2005. p. 163–86.
[8] Cha Y, Jung M. Satisfaction assessment of multi-objective schedules using neural fuzzy methodology. Int J Prod Res 2003;41(8):1831–49.
[9] G. Chen, S. Zhang, Fcamapx results for oaei 2018(2018).
[10] Chu T-C. Facility location selection using fuzzy topsis under group decisions. Int J Uncertainty Fuzziness Knowledge Based Syst 2002;10(6):687–701.
[11] Demšar J. Statistical comparisons of classifiers over multiple data sets. Journal of Machine learning research 2006;7(Jan):1–30.
[12] Du Y, Gao C, Hu Y, Mahadevan S, Deng Y. A new method of identifying influential nodes in complex networks based on topsis. Physica A 2014;399:57–69.
[13] B.C.G. E. Jimenez-Ruiz, V. Cross, Logmap family participation in the oaei 2018(2018).
[14] Edwards W, Barron FH. Smarts and smarter: improved simple methods for multiattribute utility measurement. Organ Behav Hum Decis Process 1994;60(3):306–25.
[15] Euzenat J, Shvaiko P, et al. Ontology matching, 18. Springer; 2007.
[16] D. Faria, C. Pesquita, B.S. Balasubramani, T. Tervo, D. Carriço, R. Garrilha, F.M. Couto, I.F. Cruz, Results of aml participation in oaei 2018(2018).
[17] Figueira J, Mousseau V, Roy B. Electre methods. In: Multiple criteria decision analysis: State of the art surveys. Springer; 2005. p. 133–53.
[18] Geman D, Reynolds G. Constrained restoration and the recovery of discontinuities. IEEE Transactions on Pattern Analysis & Machine Intelligence 1992(3):367–83.
[19] Geman D, Yang C. Nonlinear image recovery with half-quadratic regularization. IEEE Trans Image Process 1995;4(7):932–46.
[20] Govindan K, Kadziński M, Sivakumar R. Application of a novel promethee-based method for construction of a group compromise ranking to prioritization of green suppliers in food supply chain. Omega (Westport) 2017;71:129–45.
[21] He R, Tan T, Wang L. Robust recovery of corrupted low-rankmatrix by implicit regularizers. IEEE Trans Pattern Anal Mach Intell 2014a;36(4):770–83.

[22] He R, Zhang Y, Sun Z, Yin Q. Robust subspace clustering with complex noise. IEEE Trans Image Process 2015;24(11):4001–13.

[23] He R, Zheng W-S, Hu B-G, Kong X-W. Two-stage nonnegative sparse representation for large-scale face recognition. IEEE Trans Neural Netw Learn Syst 2013;24(1):35–46.

[24] He R, Zheng W-S, Tan T, Sun Z. Half-quadratic-based iterative minimization for robust sparse representation. IEEE Trans Pattern Anal Mach Intell 2014b;36(2):261–75.

[25] S. Hertling, H. Paulheim, Dome results for oaei 2018 (2018).

[26] Huber PJ. Robust statistics. Springer; 2011.

[27] M. Kachroudi, G. Diallo, S.B. Yahia, Kepler at oaei 2018 (2018).

[28] Kou G, Lu Y, Peng Y, Shi Y. Evaluation of classification algorithms using mcdm and rank correlation. International Journal of Information Technology & Decision Making 2012;11(01):197–225.

[29] Kou G, Peng Y, Wang G. Evaluation of clustering algorithms for financial risk analysis using mcdm methods. Inf Sci (Ny) 2014;275:1–12.

[30] A. Laadhar, F. Ghozzi, I. Megdiche, F. Ravat, O. Teste, F. Gargouri, Oaei 2018 results of pomap+ (2018).

[31] Liu H-C, Li Z, Song W, Su Q. Failure mode and effect analysis using cloud model theory and promethee method. IEEE Trans Reliab 2017;66(4):1058–72.

[32] Liu H-C, Wang L-E, Li Z, Hu Y-P. Improving risk evaluation in fmea with cloud model and hierarchical topsis method. IEEE Trans Fuzzy Syst 2018;27(1):84–95.

[33] Liu W, Pokharel PP, Príncipe JC. Correntropy: properties and applications in non-gaussian signal processing. IEEE Trans Signal Process 2007;55(11):5286–98.

[34] Mann ME, Lees JM. Robust estimation of background noise and signal detection in climatic time series. Clim Change 1996;33(3):409–45.

[35] Mohammadi M, Hofman W, Tan Y-H. Simulated annealing-based ontology matching. ACM Transactions on Management Information Systems (TMIS) 2019;10(1):3.

[36] Mustajoki J, Hämäläinen RP, Salo A. Decision support by interval smart/swing incorporating imprecision in the smart and swing methods. Decision Sciences 2005;36(2):317–39.

[37] Nikolova M, Ng MK. Analysis of half-quadratic minimization methods for signal and image recovery. SIAM Journal on Scientific computing 2005;27(3):937–66.

[38] Noghabi HS, Mohammadi M, Tan Y-H. Robust group fused lasso for multisample copy number variation detection under uncertainty. IET Syst Biol 2016;10(6):229–36.

[39] Opricovic S. Multicriteria optimization of civil engineering systems. Faculty of Civil Engineering, Belgrade 1998;2(1):5–21.

[40] Opricovic S, Tzeng G-H. Multicriteria planning of post-earthquake sustainable reconstruction. Comput-Aided Civ Infrastruct Eng 2002;17(3):211–20.

[41] Opricovic S, Tzeng G-H. Compromise solution by mcdm methods: a comparative analysis of vikor and topsis. Eur J Oper Res 2004;156(2):445–55.

[42] Peng Y, Kou G, Wang G, Shi Y. Famcdm: a fusion approach of mcdm methods to rank multiclass classification algorithms. Omega (Westport) 2011;39(6):677–89.

[43] Peng Y, Wang G, Wang H. User preferences based software defect detection algorithms selection using mcdm. Inf Sci (Ny) 2012;191:3–13.

[44] Percin S. Evaluation of third-party logistics (3pl) providers by using a two-phase ahp and topsis methodology. Benchmarking: An International Journal 2009;16(5):588–604.

[45] O.T.C.T. Philippe Roussille, Imen Megdiche, Holontology : results of the 2018 oaei evaluation campaign (2018).

[46] J. Portisch, H. Paulheim, Alod2vec matcher (2018).

[47] Rezaei J. Best-worst multi-criteria decision-making method. Omega (Westport) 2015;53:49–57.

[48] Saaty TL. A scaling method for priorities in hierarchical structures. J Math Psychol 1977;15(3):234–81.

[49] Saaty TL. Decision making for leaders: the analytic hierarchy process for decisions in a complex world. RWS publications; 1990.

[50] Shojaei P, Haeri SAS, Mohammadi S. Airports evaluation and ranking model using taguchi loss function, best-worst method and vikor technique. Journal of Air Transport Management 2018;68:4–13.

[51] K.R. Jomar da Silva, F.A. Baiao, Alin results for oaei 2018 (2018).

[52] Solimando A, Jiménez-Ruiz E, Guerrini G. Detecting and correcting conservativity principle violations in ontology-to-ontology mappings. In: International Semantic Web Conference. Springer; 2014a. p. 1–16.

[53] Solimando A, Jiménez-Ruiz E, Guerrini G. A multi-strategy approach for detecting and correcting conservativity principle violations in ontology alignments.. In: OWLED; 2014b. p. 13–24.

[54] Soylu B. Integrating prometheeii with the tchebycheff function for multi criteria decision making. International Journal of Information Technology & Decision Making 2010;9(04):525–45.

[55] Triantaphyllou E. Multi-criteria decision making methods. In: Multi-criteria decision making methods: A comparative study. Springer; 2000. p. 5–21.

[56] Tzeng G-H, Huang J-J. Multiple attribute decision making: methods and applications. Chapman and Hall/CRC; 2011.

[57] Wang H, Li H, Zhang W, Zuo J, Wang H. Maximum correntropy derivative-free robust kalman filter and smoother. IEEE Access 2018;6:70794–807.

[58] Wang H, Li H, Zhang W, Zuo J, Wang H. A unified framework for m-estimation based robust kalman smoothing. Signal Processing 2019;158:61–5.

[59] S.B.Y. Warith Eddine Djeddi, M.T. Khadir, Xmap : Results for oaei 2018(2018).