Aggregation of Multiple Judgments for Evaluating Ordered Lists

Hyun Duk Kim, ChengXiang Zhai and Jiawei Han

Department of Computer Science, University of Illinois at Urbana-Champaign,
201 N Goodwin Ave, Urbana, IL 61801, USA
hkim277@illinois.edu,czhai@cs.uiuc.edu,hanj@cs.uiuc.edu

Abstract. Many tasks (e.g., search and summarization) result in an ordered list of items. In order to evaluate such an ordered list of items, we need to compare it with an ideal ordered list created by a human expert for the same set of items. To reduce any bias, multiple human experts are often used to create multiple ideal ordered lists. An interesting challenge in such an evaluation method is thus how to aggregate these different ideal lists to compute a single score for an ordered list to be evaluated. In this paper, we propose three new methods for aggregating multiple order judgments to evaluate ordered lists: weighted correlation aggregation, rank-based aggregation, and frequent sequential pattern-based aggregation. Experiment results on ordering sentences for text summarization show that all the three new methods outperform the state of the art average correlation methods in terms of discriminativeness and robustness against noise. Among the three proposed methods, the frequent sequential pattern-based method performs the best due to the flexible modeling of agreements and disagreements among human experts at various levels of granularity.

Key words: Evaluation, Sentence ordering, Judgment aggregation, Frequent sequential pattern mining

1 Introduction

How to aggregate different human evaluators' judgments is a difficult problem in evaluation with multiple human annotations. When we evaluate the performance of a system, we often compare the output of system with a "gold standard output" created by a human evaluator; the more similar the system output is to the human-created gold standard, the better the performance of the system would be.

Unfortunately, when a task is difficult or inherently subjective to judge (as in the case of many information retrieval problems such as search and summarization), human experts may not agree with each other on the gold standard. Thus using only one single human expert to create the gold standard can be biased, and it would be necessary to have multiple experts to create a gold standard, leading naturally to multiple (gold standard) judgments, each created by a different human expert.

The research question we study in this paper is how to aggregate these multiple judgments created by multiple experts to evaluate ordered lists of items.

Evaluation of ordered lists is quite important in information retrieval because in many tasks, the output of a system is an ordered list. For example, a search engine generates a ranked list of documents, while a text summarizer generates a ranked list of extracted sentences from documents. Existing approaches to evaluation of these tasks tend to simplify the task of evaluation by not requiring human experts to create a *complete* ideal ranked list and only asking them to distinguish items that should be ranked high from those that should be ranked low. For example, in evaluating retrieval results, we often ask a human assessor to judge which document is relevant and which is non-relevant. While this alleviates the problem of disagreement among human assessors, it does not allow us to distinguish finer granularity differences in ranking, as, e.g., changing the relative order of relevant documents (or non-relevant documents) would generally not affect performance. Moreover, such a coarse judgment would not be sufficient for evaluating a task where ranking is the primary goal. For example, a common last step in most text summarization approaches is to order the extracted representative sentences from documents appropriately to generate a coherent summary. To distinguish a good ordering from a poor one, it would be better to have human experts to generate ideal orderings of the extracted sentences. Furthermore, since it is unlikely that human experts would agree on a single way of ordering a set of sentences, it is necessary to have multiple human experts to create ideal orderings, raising the challenge to aggregate these potentially different ideal orderings to evaluate any ordered list given by a system.

The current way of solving this aggregation problem is to evaluate a given ordered list from a system with each gold standard separately and then take the average of the results of individual gold standard evaluation as the overall score of the system list (see, e.g., [1]). However, this simple method does not explicitly consider and effectively reflect agreements and disagreements among different gold standards since it trusts equally every part of an ideal ordering created by every expert. Intuitively, an expert agreeing with most other experts can be trusted more, and for the same expert, different parts of an ordering may not be equally trustable. Thus ideally, we should use only the "agreed" judgments for evaluation or put different weights on different parts depending on their "degree of agreement" by all the experts.

Based on these insights, we propose three new methods for aggregating multiple order judgments to evaluate ordered lists.

The first method, called weighted correlation aggregation (WCA), models the overall agreement among the assessors and is a direct extension of the current average correlation method. In this method, we would take a *weighted* average of the correlations between the list to be evaluated and all the gold standard ordered lists, where the weight on each expert is computed based on the degree of overall agreement between this expert and all other experts.

The second method, called rank-based aggregation (RBA), models the agreement among the assessors in the rank of each item, and generates a consensus ordered list that summarizes the judgments by all the assessors. Evaluation can then be based on this single consensus ordering.

The third method, called frequent sequential pattern-based aggregation (FreSPA), models the agreement among the assessors at various levels of granularity as represented by consensus sequential patterns of various lengths. We would first identify sequential patterns and their frequencies from all the human

judgments of ordering, and then score an ordered list based on how well the list matches the frequent sequential patterns.

We compared the three proposed methods with two baseline methods on a publicly available data set for evaluating sentence ordering in text summarization. The two baseline methods are the uniform average correlation methods, using Kendall's τ and Spearman correlation, respectively; they represent the current state of the art. Experiment results show that our methods outperform both baseline methods in terms of discriminativeness and robustness against noise.

The rest of the paper is organized as follows. In Section 2, we discuss related work. We define our problem in Section 3, and present the proposed methods in Section 4. In Section 5, we evaluate our methods and discuss experiment results. We conclude our paper and discuss future work in Section 6.

2 Related Work

Most previous work on evaluating sentence ordering did not consider aggregation of human judgments [2–5]. In many cases, sentence ordering was evaluated qualitatively by human [3, 4]. For quantitative evaluation, Kendall's τ[2] and Spearman's rank correlation coefficient are often used (e.g., [2, 1, 5]). However, these measures have not considered aggregation of multiple ordering judgments made by different human evaluators or have simply taken a uniform average of the correlation values computed using multiple judgments without considering the variation of trustworthiness of human assessors, which we address in our methods.

Barzilay et al.[6] looked into the agreement among human assessors and revealed the difficulty in evaluating the task of ordering sentences in text summarization. They created multiple judgments using multiple assessors, but did not propose a method for aggregating human judgments for evaluating sentence ordering. We propose methods for automatic evaluation based on aggregation of those human generated orderings and use their data set to evaluate the proposed method.

Subjectivity in human evaluators' annotations has been studied in [7–11]. Aggregation of judgments and votes have been studied in other study fields such as logic, economy and mathematics [12–15]. These studies focus on finding the best item based on different voting results, whereas our work is to develop a method for evaluating a new ordered list based on multiple judgments.

Data fusion in information retrieval is also related to our work. The aim of data fusion is to combine many ranked results to generate one potentially better ranking. Various methods were proposed to combine ranked search lists, including, e.g., CombSUM/CombMNZ[16], ProbFuse[17], and generative model-based method[18]. Despite of its similarity to the ordering judgment aggregation, data fusion is mainly for making a new ranking and not for evaluation. In one of our proposed methods, we use a similar aggregation strategy to the one used in [16] to generate a single consensus judgment so as to convert multiple judgments into one single judgment.

Perhaps the most similar work to ours is the pyramid method for summary content evaluation [19], where agreements on content units are considered in designing a scoring method for evaluating the content of a summary, and the strategy is similar to our pattern-based aggregation. However, the pyramid method

cannot be applied to a complex task such as sentence ordering. Moreover, the pyramid method relies on manual matching of the content units, whereas our method is automatic and can be regarded as adapting the philosophy of the pyramid method for evaluating ordered lists in an automatic way.

Frequent sequential pattern mining is one of the most popular topics in data mining. Many algorithms have been developed, including the generalized sequential pattern mining algorithm (GSP)[20], sequential pattern discovery using equivalent class (SPADE)[21], PrefixSpan[22], and CloSpan[23]. In this paper, we used PrefixSpan algorithm for finding frequent sequential patterns.

3 Problem Definition

We consider the general problem of evaluating an ordered list based on multiple gold standard ordered lists created by different human assessors.

Formally, let $X = \{x_1, ..., x_k\}$ be a set of k items to be ordered. Suppose n human assessors created n (generally different) orderings of the k items in X, denoted as O_1, ..., O_n. $O_i = (x_{i_1}, ..., x_{i_k})$ is an ordering, where i_1, ..., i_k are a permutation of integers 1,..., k. Given a new ordered list to be evaluated, $O = (y_1, ..., y_k)$, our goal is to find a scoring function $s(O; O_1, ..., O_n)$ that can be used to score the ordered list O based on its consistency with the gold standard orderings $O_1, ..., O_n$.

This problem setup is general and should cover many interesting application instances. A particular case that we will use to evaluate the proposed scoring function is sentence ordering in text summarization. Virtually all the methods for automatic summarization would first extract a set of most representative sentences from documents and then order them to form a coherent summary document. To evaluate the quality of the generated summary, one aspect we must measure is the optimality of the ordering of the sentences, which presumably would directly affect the readability of the summary.

4 Methods for Ordered List Evaluation

In this section, we first present two baseline methods that represent the state of the art and then present the three new methods that we proposed.

4.1 Baseline Methods

For general ordering evaluation, Kendall's τ[2] and Spearman's rank correlation coefficient are often used. Let π and σ be two different orders, and N the length of the order. Kendall's τ is defined as $\tau = 1 - \frac{2S(\pi,\sigma)}{N(N-1)/2}$, where $S(\pi, \sigma)$ means number of discordant pairs between π and σ. For example, the two ordered lists (ABCD) and (ACDB) have two discordant pairs since the orders of BC and BD are reversed.

Spearman's rank correlation coefficient is defined as $Spearman = 1 - \frac{6\sum_{i=1}^{N}(\pi(i)-\sigma(i))^2}{N(N^2-1)}$ where $\pi(i)$ and $\sigma(i)$ mean the rank of item i in π and σ, respectively. That is, this measure uses rank difference of items in the two ordered lists. The range of both Kendall's τ and Spearman's rank correlation coefficient is [-1,1].

These two baseline methods are only able to compare a target list to one ideal ordered list. In order to use it for evaluation with multiple ideal ordered lists, in the existing work (e.g., [1]), the average of the correlation values for all the ideal ordered lists is taken as the overall score of a target list w.r.t. the multiple judgments. We thus call this baseline method *average correlation* (AC). Formally, $S_{AC}(O; O_1, ..., O_n) = \frac{1}{n} \sum_{i=1}^{n} C(O, O_i)$, where $C(O, O_i)$ is the correlation between two orderings as measured by either Kendall's τ or Spearman's rank correlation (denoted by AC-τ and AC-Sp, respectively).

The AC method gives each ideal ordered list an equal weight, thus does not address the issue of variation of trustworthiness of different assessors. Below we propose three methods to address this issue.

4.2 Weighted Correlation Aggregation (WCA)

Our first method is weighted correlation aggregation, which is a direct extension of AC by assigning different weights to different judgments, where the weight of each judgment (i.e., each assessor) is computed based on the degree of overall agreement of the assessor with other assessors. Formally, WCA is defined as:

$S_{WCA}(O; O_1, ..., O_n) = \frac{\sum_i^n w_i C(O,O_i)}{\sum_i^n w_i}$,

where the weight $w_i = \frac{1}{n-1} \sum_{j=1, j \neq i}^{n} C(O_i, O_j)$.

The two-list correlation measure $C(O, O_i)$ can be either Kendall's τ or Spearman's rank correlation, leading to two corresponding variations of the WCA method, denoted by WCA-τ and WCA-Sp, respectively.

In WCA, the weight on each assessor is based on the *overall* agreement between the assessor and other assessors. However, it may be desirable to model the agreement of assessors at a finer granularity level, which motivates us to propose the following rank-based aggregation method.

4.3 Rank-based Aggregation (RBA)

In this method, we model the agreement of assessors at the level of the rank of each item and aggregate all the ranks to generate a consensus ordered list which can summarize the overall judgments. Specifically, the consensus ordered list is generated by ordering items based on their combined ranking scores:

Combined Ranking Score of $x_i = \sum_{j=1}^{n} Rank_j(x_i)$,

where $Rank_j(x_i)$ is the rank of x_i in O_j.

A combined ranking score is lower (thus the item would be ranked higher in the consensus list) when it was highly ranked in more ideal ordered lists. Since small errors can be compensated by other experts in the summing process, this method can be more robust against noise.

After obtaining the consensus list, we can use either Kendall's τ or Spearman's rank correlation to compute the correlation of a target list with the consensus list, leading again to two variations of the RBA method, denoted by RBA-τ and RBA-Sp, respectively.

4.4 Frequent Sequential Pattern-based Aggregation (FreSPA)

In this method, we attempt to model the agreement at various levels of granularity based on frequent sequential patterns. A sequential pattern is a subsequence

of items in an ordering O_i possibly with gaps. For example, if an ordering is ABCD, both "BC" and "BD" are potential sequential patterns. Intuitively, a sequential pattern of an ordering captures the desired ordering of a subset of items. Thus if a sequential pattern has high frequency in all the judgments $O_1, ..., O_n$, it would mean that it has captured an ordering of a subset of items agreed by most assessors. We can thus use these frequent sequential patterns to score a target ordered list.

Formally, we denote a sequential pattern by pat_i, its length by $seqLen_i$, and its support (i.e., frequency) by sup_i. We would be especially interested in frequent sequential patterns (with a support at least $minSup$) in the ideal ordered lists $O_1, ..., O_n$, as they capture the agreed partial ordering of a subset of items.

Furthermore, there will be a trade-off in the length of the patterns. On the one hand, longer patterns are more useful as they give ordering information for more items, but on the other hand, they are also less likely agreed by more assessors. To retain the flexibility, we introduce two parameters, $minLen$ and $maxLen$, to restrict the lengths of patterns to be considered. If we set $minLen = 2, maxLen = 2$, we will only find length-2 pair-wise sequential patterns. If we restrict the $maxLen$ to 3, we only find frequent sequential patterns whose length is 2 or 3. Naturally, both $minLen$ and $maxLen$ must be between 2 and k.

Let P denote all the sequential patterns in $O_1, ..., O_n$ that satisfy our constraints. The FreSPA score for a target order which we want to evaluate, $O = y_1, y_2, ..., y_k$, is defined as follows:

$$S_{FreSPA}(O; O_1, ...O_n) = \frac{\sum_{pat_i \in O}(1+wLen*(seqLen_i-1))*(1+wSup*(sup_i-1))}{\sum_{pat_i \in P}(1+wLen*(seqLen_i-1))*(1+wSup*(sup_i-1))}$$

where $wLen$ and $wSup$ are two weighting parameters on the length and support (i.e., frequency) of a pattern. Clearly, $S_{FreSPA} \in [0, 1]$.

$wLen$ decides how much more weight we will put on a longer pattern. When $wLen = 0$, patterns with different lengths would have the same score contribution, whereas when $wLen = 1$, patterns m times longer would have m times more score contributions. $wSup$ decides how much more weight we will put on patterns with higher supports. When $wSup = 0$, all patterns have the same score contribution regardless of support, and when $wSup = 1$, patterns with m times higher support would have m times higher score contributions.

These parameters allow us to tune FreSPA based on the characteristics of any given set of judgments (e.g., level of consensus, amount of noisy judgments). For example, if all the judgments mostly agree with each other, we may trust longer patterns more. These parameters can be set empirically based on optimizing some effectiveness measure on the given set of judgments. Note that since such tuning can always be done before we apply FreSPA to evaluate any given ordered lists, the parameter tuning is not a problem; indeed, they provide the needed flexibility to adjust to different evaluation data set.

A main computational challenge is to find all the frequent patterns quickly. This can be done by using an efficient frequent sequential pattern mining algorithm. Many such algorithms have been proposed in the data mining community (see Section 2 for a discussion about them). In our experiments, we used the PrefixSpan algorithm [22] because it is very efficient and generates all sub-patterns. Since the number of human assessors is usually not very large, in general, the scalability of FreSPA is not a concern.

5 Experiment Results

In this section, we evaluate the proposed methods using the task of ordering sentences for text summarization. We chose this task because of the availability of a data set with multiple human judgments for evaluating our methods.

5.1 Data Set

We use the data set created by researchers for studying sentence ordering in summarization [6, 24]. The data set is publicly available on the web[1]. The data is created by asking human assessors to generate an ideal ordering of a set of sentences extracted from documents to form a coherent summary. The detailed instructions for the assessors are also published on the web [2]. There are 10 sentence sets. On average, each set of sentences has 8.8 sentences and is judged by 10.4 assessors.

5.2 Metric

Given a set of multiple judgments of ordering, how do we know whether one aggregation method is effective than another? We propose to measure the effectiveness of an aggregation method based on its "Evaluation Discriminativeness" (ED), which is defined as the score difference between a good order and a bad order of items.

Intuitively, a good aggregation method should give a high score to a good order and a low score to a bad order. Thus, the difference of the scores given to a good order and a bad order can indicate how well a method can distinguish a good order from a bad order; the higher the ED value is, the more discriminative the method is.

The remaining question is: how do we know which is a good order and which is a bad order? The fact that we have multiple gold standard judgments makes it tricky to answer this question. To avoid bias, we take the judgment from each assessor as a good order and take its reverse order as a bad order. We then compute the average ED over all the assessors. Since a gold standard order created by a human assessor can be assumed to be a good order, it would be reasonable to assume that its reverse is a bad order.

Formally, the ED value of an aggregation-based scoring method S is defined as:

$$ED = \frac{1}{n}\sum_{i}^{n}(S(O_i; O_1, ..., O_{i-1}, O_{i+1}, ..., O_n) - S(O_{i\,R}; O_1, ..., O_{i-1}, O_{i+1}, ..., O_n))$$

where $O_{i\,R}$ is reverse order of O_i. When computing the ED value on the judgment created by one assessor, we use all the other assessors' judgments as gold standards. This is essentially similar to leave-one-out cross-validation. The final overall ED score of method S is the average of the ED scores on all the assessors. Through optimizing the ED value, we can tune any parameters of method S to maximize its discriminativeness; the tuned parameter setting can then be used to evaluate any new ordered lists generated by a system.

[1] `http://www1.cs.columbia.edu/~noemie/ordering/`
[2] `http://www1.cs.columbia.edu/~noemie/ordering/experiments/`

Since the ED measure requires all the measures to have the same score range, we normalize both Kendall's τ and Spearman's rank correlation using the min-max normalization method, $(x + 1)/2$, so that the scores would be in [0,1], the range of the FreSPA score.

In addition to the discriminativeness, we also examine the robustness to noisy judgments of an aggregation-based evaluation method. Since the judgment of an optimal order in our problem setup is inherently subjective, there may be noisy judgments in our gold standard in the sense that some judgments may not be reliable, which can be caused by, e.g., careless judgment or biased judgment by an assessor. To compare their robustness, we also compare different methods by adding different amounts of random judgments into our data set.

5.3 Basic Comparison

We first present results from comparing all the variants of the proposed three methods (i.e., WCA-τ, WCA-Sp, RBA-τ, RBA-Sp, and FreSPA) with the two versions of the baseline method (i.e., AC-τ and AC-Sp) on the original data set.

The FreSPA method has several parameters to set. For this comparison, we set $wLen = 1.0$, $wSup = 1.0$, and $minSup = 0.75$ (a pattern must occur in at least 75% of the judgments), and used patterns of all lengths (i.e., $minLen = 2$ and $maxLen = k$).

The results are shown in the first row of Table 1. We see that all the three proposed methods outperform the corresponding baseline methods. Among the three proposed methods, FreSPA is the best, followed by RBA, which outperforms WCA. Since both FreSPA and RBA model agreement of assessors at a finer granularity level than WCA which models the overall agreement, this suggests that it is more effective to aggregate judgments at finer granularity level. Furthermore, the fact that FreSPA outperforms RBA suggests that it may be beneficial to model agreement at various levels of granularity. Note that the parameter setting we used here for FreSPA is not the best configuration for this method, and it is possible to further tune these parameters to achieve better ED as will be shown later when we analyze the influence of the parameters on the performance of FreSPA. We also see that Spearman generally performs better than Kendall's τ.

Table 1. ED of different methods on original and noisy data sets

Noise ratio	AC-τ	AC-Sp	WCA-τ	WCA-Sp	RBA-τ	RBA-Sp	FreSPA
0	0.454	0.542	0.459	0.549	0.564	0.676	0.722
0.25	0.300	0.354	0.357	0.435	0.460	0.564	0.656
0.5	0.220	0.264	0.236	0.292	0.380	0.468	0.551
0.75	0.167	0.202	0.173	0.215	0.358	0.433	0.510
1	0.113	0.136	0.133	0.159	0.307	0.379	0.395
Max-Min	0.341	0.406	0.326	0.389	0.257	0.297	0.327
% Degradation	75.1%	74.9%	71.0%	70.9%	45.6%	43.9%	45.3%

5.4 Comparison on Noisy Data

To test the robustness of these methods to noise, we repeated the comparison presented earlier by systematically adding some random orders into the gold standard judgment set. Specifically, we use a "noise ratio" parameter r to control the amount of noise to be added and add nr random orders to a data set with n original judgments. For example, if there are 10 human orderings and r is 0.5, we would add 5 random orders to the data set.

Table1 shows the ED values of all the methods with different noise ratios. Overall, the conclusions we made earlier on the original data set are all confirmed here. In particular, for all levels of noise ratios, the proposed methods outperform the baselines, FreSPA performs better than RBA which outperforms WCA, and Spearman performs better than Kendall's τ.

In the last two rows, we show the absolute degradation and relative percentage of degradation of ED from the original data set when we add maximum amount of noise ($r = 1.0$). We see that our proposed methods have less degradation than their corresponding baseline methods because of modeling agreement among the assessors. This is most clearly seen from the improvement of the two WCA variants over their corresponding baselines as their only difference is that in WCA, we will be able to assign lower weights to noisy orders as they are less likely correlated with other judgments. Moreover, RBA and FreSPA have significantly less degradation than WCA, indicating again the benefit of modeling agreement at finer granularity level. Indeed, when we use units such as ranks of items and frequent sequential patterns to model agreement, we can expect to eliminate most noise automatically due to their lack of agreement from other assessors. In particular, the minimum support threshold in FreSPA helps ensure that we only use reliable judgments.

5.5 Parameter Setting of FreSPA

While the WCA and RBA have no parameter to tune, FreSPA has several parameters to set, allowing it to adapt to special characteristics of a set of judgments. We now look into the influence of the setting of these parameters on the ED of FreSPA by varying the parameter values on both the original and noisy data sets. Unless otherwise stated, the parameters that are not varied are set to their default values, which we used for comparing different methods (i.e., $wLen = wSup = 1.0$, $minLen = 2$, $maxLen = k$, $minSupp = 0.75$).

Table 2. ED with different wLen and wSup

wLen	ED (original)	ED (noisy)	wSup	ED (original)	ED (noisy)
0	**0.7301**	0.5255	0	0.7106	0.5704
0.25	0.7261	0.5517	0.25	0.7186	0.5145
0.5	0.7239	0.4736	0.5	0.7203	0.5504
0.75	0.7225	0.5327	0.75	0.7211	0.5141
1	0.7215	0.5482	1	0.7215	**0.6019**
5	0.7177	0.5642	5	0.7227	0.5694
10	0.7169	0.5828	10	0.7229	0.5616
20	0.7165	0.5874	20	0.7230	0.5275
50	0.7163	**0.6120**	50	0.7230	0.5361
100	0.7162	0.5403	100	**0.7230**	0.5130

We first show the results from varying $wLen$ and $wSup$ in Table2. We see that the optimal settings of these two parameters differ between the original data set and the noisy data set. In particular, for the original (clean) data set, it is better to set $wLen = 0$, meaning that all patterns are equally important. However, for the noisy data, it is important to set $wLen$ to a large value so as to give more weight to longer patterns, which makes sense as when there is noise, longer patterns would be more trustable (since the high frequency of a short pattern is likely due to chance). Similarly, for the clean data, it is better to set $wSup$ to a large value, which means we can trust the support computed over all the judgments, whereas on the noisy data, setting it to a moderate value (around 1.0) performs the best, which means that we cannot entirely trust the voting from all the judgments.

Table 3. ED with different maxLen and minLen ratios

maxLen/k (minLen=2)	ED (original)	ED (noisy)	minLen/k (maxLen=k)	ED (original)	ED (noisy)
0	0.7493	0.5265	0	0.7215	0.5841
0.25	0.5818	0.4362	0.25	0.7215	0.5361
0.5	0.7267	0.5331	0.5	0.4137	0.0707
0.75	0.7215	0.5320	0.75	0.0824	0.0000
1	0.7215	0.5303	1	0.0000	0.0000

Next, in Table3, we show the results from varying the pattern length parameters $minLen$ and $maxLen$ on both the original and noisy data sets. Since different sentence sets have different numbers of sentences (i.e., different k), we vary these two parameters based on their ratio to the total number of items to order k, which is also the maximum value that $minLen$ and $maxLen$ can take. A ratio of 0 means that $minLen$ (or $maxLen$) is set to 2 since their minimum value is 2. This time, the difference between the original and noisy data sets appears to be less clear, and it appears that using all the patterns (i.e., $maxLen = k$ and $minLen = 2$) is desirable. Also, it is clear that if we use only long patterns, as we would expect, the performance is very poor. However, $maxLen = 2$ seems to be an "outlier" when the performance is actually very good; further analysis would be needed to understand why.

Finally, we look at $minSup$, which controls the threshold for support, i.e., the frequency of a pattern. Intuitively, FreSPA would be more reliable with a higher $minSup$, which would demand more consensus by the assessors unless there are noisy judgments. To verify this intuition, we varied this parameter on both the original and noisy data sets. The results are shown in Table 4. The results confirm our intuition as on the original data set, the optimal value of $minSup$ is 1.0, meaning that it is important to require a pattern to be agreed by every assessor. When $minSup = 1$, we probably would use many short patterns as they are more likely to be agreed by all assessors. However, on the noisy data set, the optimal value is 0.75, which means that we should require only 75% assessors to agree on a pattern, which is reasonable given that some judgments are noise.

Table 4. ED with different minimum support

minSup	ED (original)	ED (noisy)
0	0.1741	0.0815
0.25	0.3678	0.2639
0.5	0.5328	0.4040
0.75	0.7215	0.5855
1	0.8985	0.0230

6 Conclusions

How to aggregate multiple human judgments to evaluate an ordered list is an important challenge in evaluating many information retrieval tasks. Previous work has not addressed well the variation of trustworthiness of different assessors. In this paper, we proposed three new methods to better address this issue, including the weighted correlation aggregation (WCA), rank-based aggregation (RBA), and frequent sequential pattern-based aggregation (FreSPA). Evaluation using a sentence ordering data set shows that all the three new methods outperform the state of the art average correlation methods in terms of discriminativeness and robustness against noise. Among the three proposed methods, FreSPA performs the best due to the flexible modeling of agreements and disagreements among human experts at various levels of granularity. Moreover, RBA and FreSPA are more effective and more robust than WCA due to modeling of agreement at a finer granularity level.

All the proposed methods are quite general and can be applied to any task where ordering needs to be evaluated with multiple human judgments. The general idea of the FreSPA method can also be extended to evaluate other non-ordering tasks. For example, by finding frequent patterns instead of finding frequent *sequential* patterns, we can adapt FreSPA to evaluate the content of a summary for summarization evaluation. The idea would be very similar to the pyramid method[19], but the modified FreSPA can be expected to give more flexibility and better performance with minimum support. These will be interesting directions for future work.

Acknowledgments. This material is based upon work supported by the National Science Foundation under Grant Numbers IIS-0347933, IIS-0713581, and IIS-0713571.

References

1. Lapata, M.: Probabilistic text structuring: experiments with sentence ordering. In: Proceedings of ACL '03, Morristown, NJ, USA, Association for Computational Linguistics (2003) 545–552
2. Lapata, M.: Automatic evaluation of information ordering: Kendall's tau. Comput. Linguist. **32**(4) (2006) 471–484
3. Okazaki, N., Matsuo, Y., Ishizuka, M.: Improving chronological sentence ordering by precedence relation. In: Proceedings of COLING '04, Morristown, NJ, USA, Association for Computational Linguistics (2004) 750
4. Bollegala, D., Okazaki, N., Ishizuka, M.: A bottom-up approach to sentence ordering for multi-document summarization. In: Proceedings of ACL '06, Morristown, NJ, USA, Association for Computational Linguistics (2006) 385–392

5. Bollegala, D., Okazaki, N., Ishizuka, M.: A machine learning approach to sentence ordering for multidocument summarization and its evaluation. In: Proceedings of IJCNLP '05. Volume 3651 of Lecture Notes in Computer Science., Springer (2005) 624–635
6. Barzilay, R., Elhadad, N., McKeown, K.R.: Inferring strategies for sentence ordering in multidocument news summarization. In: Journal of Artificial Intelligence Research. Volume 17. (2002) 35–55
7. Reidsma, D., op den Akker, R.: Exploiting 'subjective' annotations. In: Proceedings of HumanJudge '08, Morristown, NJ, USA, Association for Computational Linguistics (2008) 8–16
8. Wilson, T.: Annotating subjective content in meetings. In: Proceedings of LREC '08, Marrakech, Morocco, European Language Resources Association (ELRA) (2008) http://www.lrec-conf.org/proceedings/lrec2008/.
9. Beigman Klebanov, B., Beigman, E., Diermeier, D.: Analyzing disagreements. In: Proceedings of HumanJudge '08, Manchester, UK, International Committee on Computational Linguistics (2008) 2–7
10. Passonneu, R., Lippincott, T., Yano, T., Klavans, J.: Relation between agreement measures on human labeling and machine learning performance: Results from an art history domain. In: Proceedings of LREC '08, Marrakech, Morocco (2008)
11. Wiebe, J.M., Bruce, R.F., O'Hara, T.P.: Development and use of a gold-standard data set for subjectivity classifications. In: Proceedings of ACL '99, Morristown, NJ, USA, Association for Computational Linguistics (1999) 246–253
12. Lang, J.: Vote and aggregation in combinatorial domains with structured preferences. In: Proceedings of IJCAI'07, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (2007) 1366–1371
13. Dietrich, F., List, C.: Judgment aggregation by quota rules. Public Economics 0501005, EconWPA (2005)
14. Hartmann, S., Sprenger, J.: Judgment aggregation and the problem of tracking the truth. (2008)
15. Drissi, M., Truchon, M.: Maximum likelihood approach to vote aggregation with variable probabilities. Technical report (2002)
16. Fox, E.A., Shaw, J.A.: Combination of multiple searches. In: TREC. (1993) 243–252
17. Lillis, D., Toolan, F., Collier, R., Dunnion, J.: Probfuse: a probabilistic approach to data fusion. In: Proceedings of SIGIR '06, New York, NY, USA, ACM (2006) 139–146
18. Efron, M.: Generative model-based metasearch for data fusion in information retrieval. In: Proceedings of JCDL '09, New York, NY, USA, ACM (2009) 153–162
19. Nenkova, A., Passonneau, R., McKeown, K.: The pyramid method: Incorporating human content selection variation in summarization evaluation. ACM Trans. Speech Lang. Process. **4**(2) (2007) 4
20. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalizations and performance improvements. In: Proceedings of EDBT '96, London, UK, Springer-Verlag (1996) 3–17
21. Zaki, M.J.: Spade: An efficient algorithm for mining frequent sequences. Mach. Learn. **42**(1-2) (2001) 31–60
22. Pei, J., Han, J., Mortazavi-asl, B., Pinto, H., Chen, Q., Dayal, U., chun Hsu, M.: Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In: Proceedings of ICDE '01, Washington, DC, USA, IEEE Computer Society (2001) 215
23. Yan, X., Han, J., Afshar, R.: Clospan: Mining closed sequential patterns in large datasets. In: Proceedings of SDM '03. (2003) 166–177
24. Barzilay, R., Elhadad, N., McKeown, K.R.: Sentence ordering in multidocument summarization. In: Proceedings of HLT '01, Morristown, NJ, USA, Association for Computational Linguistics (2001) 1–7