s-AWARE: supervised measure-based methods for crowd-assessors combination

Marco Ferrante[1], Nicola Ferro[2], and Luca Piazzon[2]

[1] Department of Mathematics "Tullio Levi-Civita", University of Padua, Italy
ferrante@math.unipd.it
[2] Department of Information Engineering, University of Padua, Italy
{ferro,piazzonl}@dei.unipd.it

Abstract. Ground-truth creation is one of the most demanding activities in terms of time, effort, and resources needed for creating an experimental collection. For this reason, crowdsourcing has emerged as a viable option to reduce the costs and time invested in it.
An effective assessor merging methodology is crucial to guarantee a good ground-truth quality. The classical approach involve the aggregation of labels from multiple assessors using some voting and/or classification methods. Recently, *Assessor-driven Weighted Averages for Retrieval Evaluation (AWARE)* has been proposed as an unsupervised alternative, which optimizes the final evaluation measure, rather than the labels, computed from multiple judgments.
In this paper, we propose s-AWARE, a supervised version of AWARE. We tested s-AWARE against a range of state-of-the-art methods and the unsupervised AWARE on several TREC collections. We analysed how the performance of these methods changes by increasing assessors' judgement sparsity, highlighting that s-AWARE is an effective approach in a real scenario.

Keywords: crowdsourcing, ground-truth, assessor merging, AWARE

1 Introduction

System-oriented evaluation is based on the use of experimental collections consisting of document corpora, topics, and relevance judgements, defining which documents are relevant for which topics. Obtaining relevance judgments and creating the ground-truth is a human-based activity and it is one of the most demanding tasks in preparing an experimental collection. Traditionally, it has been performed by relying on expert assessors [11], being quite onerous in terms of time and costs.

Therefore, a more recent approach to ground-truth creation relies on crowdsourcing [2, 3]. Multiple judgements are collected for each document from many crowd-assessors, possibly less qualified than the experts but cheaper, leveraging on the larger number of assessors to shorten the overall task execution time. The multiple judgments by crowd-assessors are then merged together, with the overall objective to achieve an assessment quality comparable to the one of traditional expert assessors. Several studies, e.g. [4], have shown that crowd-assessors often agree with experts, in particular when it comes to relevant documents [5].

Traditional approaches, like *Majority Vote (MV)* [14] or *Expectation Maximization (EM)* [8], merge multiple labels by the different crowd-assessors into a final label which is used as the relevance judgement to compute performance measures. However, a labelling error at the ground-truth level may have a different impact on different measures. For example, suppose that in the top-five documents one is actually relevant while another one is mislabelled as relevant; precision at five will have the same value, independently of the rank position of the mislabelled document; on the other hand, *Average Precision (AP)* will have different values depending on the rank position of the mislabelled document. Therefore, the same error may have different effects on different measures and also on different runs for the same measure, since different runs may rank the mislabelled document differently. To overcome these issues, Ferrante et al. [6] proposed *Assessor-driven Weighted Averages for Retrieval Evaluation (AWARE)* which, differently from traditional approaches, computes performance measures based on each crowd-assessor judgements and then merges these crowd-measures into a final weighted measure, optimizing the merging process to the considered measures and runs.

While AWARE adopts an unsupervised approach to determine the weights to be used to merge the crowd-measures, in this paper we propose a supervised extension of AWARE, that we call s-AWARE. We evaluate our s-AWARE against unsupervised AWARE and state-of-the-art supervised and unsupervised methods by using the TREC 2012 Crowdsourcing track [12] and the TREC 2017 Common Core track [1] datasets.

The paper is organized as follows: Section 2 presents some related work; Section 3 explains the s-AWARE methodology; Section 4 describes the experiments and the evaluation results; Section 5 draws some conclusions and outlooks for future work.

2 Related Works

The most common approach, still very effective, to crowd-assessor merging is *Majority Vote (MV)* [14]: it assigns to each document the most popular judgement among those expressed by crowd-assessors; to deal with variable quality workers, several weighted versions of MV have been proposed, e.g. [15, 14].
Expectation Maximization (EM) [8] addresses the problem in a probabilistic way, by iteratively estimating the probability of relevance of each document and then by assigning it the most probable judgement. Several versions of EM algorithms have been proposed, optimizing whether the document relevance probability in an unsupervised [8] or semi-supervised way [13]. Georgescu and Zhu [7] proposed an EM method for optimizing the assessors' reliability used to dinamically merge crowd judgements. Whiting et al. [18] proposed a network based approach to estimate the assessor's trustworthiness, using a modified version of PageRank.

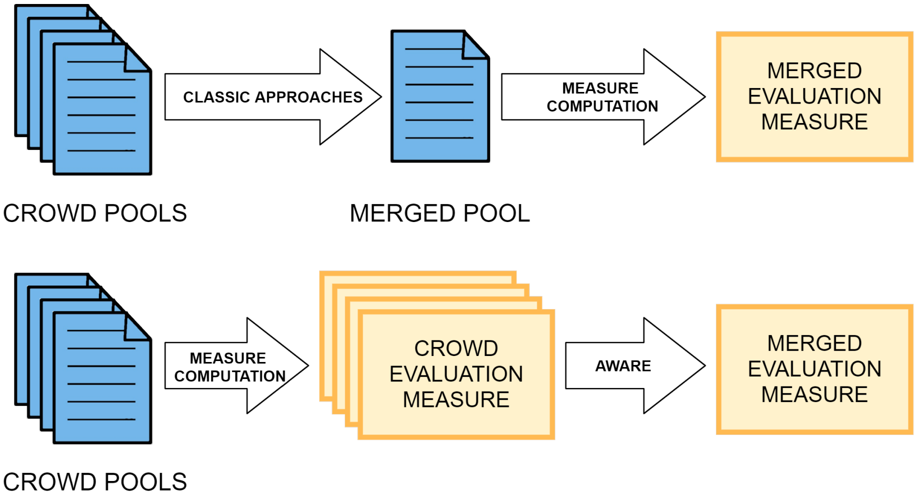


Fig. 1: Traditional vs AWARE approach.

Nellapati et al. [10] developed a mixed method, combining expert supervision, machine learning algorithms and automatic error correction.

As shown in Figure 1, all the above methods end up by selecting an optimal label, according to some criterion, among those assigned by crowd-assessors and producing a single merged pool then used to compute performance measures. However, different evaluation measures can be unfairly affected in by mislabelled documents. Therefore, *Assessor-driven Weighted Averages for Retrieval Evaluation (AWARE)* [6] directly computes the performance of a system on the judgements given by every crowd-assessor and then combine the obtained measures by weighting each assessor on the basis of her/his estimated accuracy:

$$aware\_\mu(r_t) = \sum_{k=1}^{m} \mu\left(\hat{r}_t^k\right) \frac{a_k(t)}{\sum_{h=1}^{m} a_h(t)}$$

where m is the number of crowd-assessors to merge, $\mu\left(\hat{r}_t^k\right)$ is the value of the performance measure computed on run r for topic t according to the k-th crowd-assessor, and a_k is the accuracy of the k-th crowd-assessor.

AWARE adopts an unsupervised approach to compute the a_k accuracy scores: the more a crowd-assessor is "far way" from three random assessors (uniform, over-estimating relevance, under-estimating relevance), the more accurate the crowd-assessor is.

We will refer to this unsupervised version of AWARE as u-AWARE when needed to distinguish it from the supervised version proposed in this paper.

3 s-AWARE Methodology

s-AWARE adopts a supervised approach where the more a crowd-assessor is "close" to the gold standard, the better is her/his accuracy.

Given a set of systems S and a set of topics T, let M_k be the k-th crowd-measure, i.e. the $|T| \times |S|$ matrix containing the performance scores computed on the judgments of the k-th crowd-assessor; let M^* be the performance measure corresponding to the gold standard. We consider two alternatives to quantify the "closeness"C_k to the gold standard[3]:

- *Measure closeness*: we consider the *Root Mean Square Error (RMSE)* between the crowd-measure and the gold standard one

$$C_k = RMSE\left(\overline{M}_k(\cdot, S) - \overline{M}^*(\cdot, S)\right) = \sqrt{\sum_{s=1}^{|S|} \frac{\left(\overline{M}_k(\cdot, s) - \overline{M}^*(\cdot, s)\right)^2}{\mid S \mid}}$$

  where $\overline{M}(\cdot, s)$ indicates the average measure by topic
- *Ranking of Systems closeness*: we use the Kendall's τ correlation between the ranking of systems using the crowd-measure and the gold standard one

$$C_k = \tau\left(\overline{M}_k(\cdot, S), \overline{M}^*(\cdot, S)\right) = \frac{A - D}{\mid S \mid (\mid S \mid -1)/2}$$

  where A is the number of system pairs ranked in the same order in $\overline{M}_k(\cdot, S)$ and $\overline{M}^*(\cdot, S)$, and D is the number of discordant pairs.

All the "closenesses"C_k are then normalized in the [0,1] range, setting normalized C_k equal to 1 with gold standard behaviour (RMSE equal to 0 or Kendall's τ equal to 1).

Finally, to further emphasize the "closeness", accuracy scores a_k are computed as: the original normalized C_k, the squared C_k and the cubed C_k. Algorithm 1 summarizes the accuracy computation process.

4 Evaluation

4.1 Experimental Setup

We compared s-AWARE approaches against the following baselines:

- unsupervised
  - Majority Vote (`mv`) [14];

[3] The original AWARE methodology considered additional ways to quantify "closeness", i.e. Frobenious norm, *Kullback-Leibler Divergence (KLD)*, and *AP Correlation (APC)*. Here, we focus on the two approaches which produced the best and most stable results across different configurations.

Algorithm 1: s-AWARE accuracy computation.

```
Data: $T$ training topic set; $\hat{r}_t^k \ \forall t \in T$ ground truth generated by assessor k; $\hat{r}_t \ \forall t \in T$
      experts ground truth
Result: $a_k$ accuracy score for assessor k
$M_k \leftarrow$ compute $\mu(\cdot)$ on $\hat{r}_t^k$;                         // assessor measures
$M^* \leftarrow$ compute $\mu(\cdot)$ on $\hat{r}_t$;                           // gold measures
if $RMSE$ then
    $C_k = RMSE\left(\overline{M}_k(\cdot, S) - \overline{M}^*(\cdot, S)\right)$ ;    // Closeness computation
    $w_k = 1 - C_k$ ;                                               // [0,1] normalization
else if $Kendall\ Tau$ then
    $C_k = \tau\left(\overline{M}_k(\cdot, S), \overline{M}^*(\cdot, S)\right)$ ;        // Closeness computation
    $w_k = \mid C_k \mid$ ;                                              // [0,1] normalization
end
if squared closeness then $a_k = w_k^2$;
else if cubed closeness then $a_k = w_k^3$;
else $a_k = w_k$;
```

    - Expectation Maximization with MV seeding (`emmv`) [8];
    - u-AWARE with uniform accuracy scores (`uniform`);
    - u-AWARE with squared distance from random assessors (`unsup_rmse_tpc`, `unsup_tau_tpc`), using RMSE and Kendall's τ, respectively, for "closeness" computation;
- supervised or semi-supervised
    - supervised EM method (hard labels, PN discrimination, no boost version) (`emGZ`) [7];
    - semi-supervised EM (`emsemi`) [13], using the same training-test proportion of s-AWARE.

We used *Average Precision (AP)* as performance measure. To evaluate the different approaches, as done in the TREC 2012 Crowdsourcing track, we used the *AP Correlation (APC)* [19] between the ranking of systems induced by each merging approach and the gold standard.

We used the TREC 2012 Crowdsourcing track [12] data where participating groups submitted 31 pools for 10 topics; these 10 topics were used in TREC 08 Adhoc track (`T08`) [17], consisting of 129 runs, and TREC 13 Robust track (`T13`) [16], consisting of 110 runs. We also used a portion of real crowd-sourced data from the TREC 2017 Common Core track dataset (`T26`) [1], consisting of 75 runs and 50 topics; Inel et al. [9] gathered relevance judgments by 406 crowd-assessors, considering a subcorpus of NYTimes containing short documents ($\leq$ 1000 words) and providing 7 judgments for each (topic, document) pair. In both cases, we used the original NIST judgments as gold standard.

Since the first aim of crowd-sourcing is to save time and costs, relying on a large expert-assessors training set is not feasible in a real scenario. For this reason, we considered an extremely challenging 30%-70% split between training and test, repeated 100 times, i.e. we used 3 topics as training and 7 topics as test for `T08` and `T13` and 15 topics as training and 35 topics as test for `T26`. In all the cases, we considered k-tuples from 2 to 7 crowd-assessors and for each k-tuple size we repeated all the computations 100 times, for validation purposes.

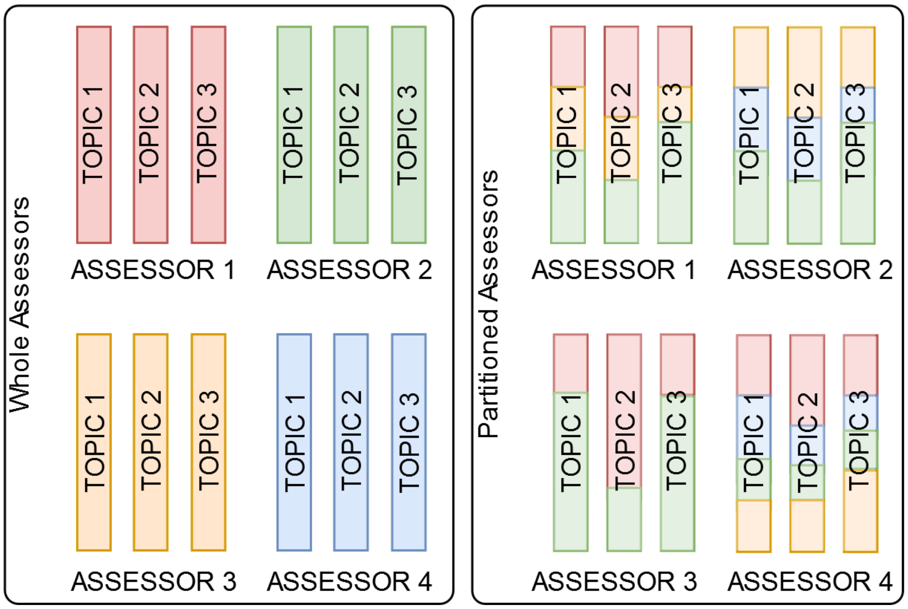


Fig. 2: Crowd-assessors experimental assembling

Since s-AWARE trains on a set of topics and `emsemi` trains on a partition of the documents for each topic, the evaluation is computed on the intersection of the two test sets (i.e. 70% of the documents from 70% of the topics)

We explore two configurations of crowd-assessors, that we call *Whole Assessors* (Figure 2 on the left) and *Partitioned Assessors* (Figure 2 on the right).

In the *Whole Assessors* case, each crowd-assessor judges completely all the topics; this is the ideal and most favourable condition for supervised and semi-supervised approaches because the crowd-assessors we learn from in the training phase exactly match those we are evaluated against in the test phase. This configuration is possible only for the `T08` and `T13` tracks, since in the TREC Crodwsourcing 2012 track each participating group judged all the topics, but not for the `T26` track.

In the *Partitioned Assessors* case, each crowd-assessor judges just some documents of a topic and she/he possibly does not judge all the topics. Therefore, the final set of judgements for each topic is assembled by combining judgements coming from more assessors, in different proportions from topic to topics, and also using different assessors for different topics. This is a more frequent case in real crowd-sourcing scenarios and it is more challenging for supervised and semi-supervised approaches since what they learn from in the training phase only partially matches what they are evaluated against in the testing phase. This is exactly the condition of the `T26` tracks, where more crowd-assessors contribute to the judgments of each topic. We also simulated this configuration on the `T08` and T13 tracks, by assembling the judgments coming from more participants into each topic.

To ease the reproducibility of the experiments, the source code is available at: https://bitbucket.org/Lucapiaz/clef2020_saware/.

4.2 Experimental Results

Table 1 reports the comparison among the different approaches in terms of AP Correlation on different tracks and for various k-tuple sizes. Baseline approaches are in blue, u-AWARE ones in green, and s-AWARE ones in orange; the darker the color, the higher the performance in terms of *AP Correlation (APC)*; best performing approaches are in bold.

In the *Whole Assessors* case, the s-AWARE sup_tau_cubed approach constantly outperforms all the other approaches for all the k-tuple sizes on both T08 and T13. This supports the idea that the *Whole Assessors* case is the most favorable to supervised approaches, since we find the same crowd-assessors both in the training and test sets and crowd-assessor judge whole topics. However, the same does not happen for the supervised and semi-supervised baseslines – emGZ and emsemi – which have lower performance than all the s-AWARE approaches and most of the unsupervised approaches, especially emGZ on T13. We hypothesize that this is due to s-AWARE approaches being much more effective at exploiting even a small training set (remember we use 30% data for training and 70% for testing). When it comes to s-AWARE alternatives, we can observe as Kendall's τ performs better than RMSE as "closeness" quantification and that the more sharp cubed weighting typically gains some more performance. We can also note how u-AWARE approaches have good performance too, typically better than state-of-the-art baselines, confirming the previous findings by [6]. Finally, we can observe as the performance of all the approaches tend to increase as the k-tuple size increases.

In the *Partitioned Assessors* case, we can observe that on T08 and T13 u-AWARE performs generally better than s-AWARE and the state-of-the-art baselines. This supports the idea that the *Partitioned Assessors* case is the most favorable to unsupervised approaches, since the training phase reflects less what happens in the test phase; k-tuples size $2, 3, 4$ on T13 are an exception, since s-AWARE outperforms all the other approaches. In general, we can observe that s-AWARE still performs remarkably better than the supervised and semi-supervised baselines – emGZ and emsemi – and better than the other unsupervised baselines. In a sense, this turns out to be a "duel" all internal to the AWARE family, which seems to better adapt to this fragmented case. This is further highlighted by the case of T26, where s-AWARE always outperforms all the other approaches. We hypothesize this is due to the fact that T08 and 13 partitioned assessor are a bit more fragmented, i.e. smaller pieces from more crowd-assessors, than the T26 ones, where there is a bunch of crowd-assessors who judge a large part of several topics. Therefore, the gap between the training and test phases is slightly smaller in this case and s-AWARE better exploit the additional information available. As in the previous *Whole Assessors* case, cubed

Table 1: Baseline approaches in blue, u-AWARE ones in green, s-AWARE ones in orange. The darker the color, the higher the performance in terms of *AP Correlation (APC)*. Best performing approaches are in bold.

		sup_rmse	sup_tau	sup_rmse_squared	sup_tau_squared	sup_rmse_cubed	sup_tau_cubed	unsup_rmse_tpc	unsup_tau_tpc	uniform	mv	emmv	emGZ	emsemi
T08-whole	k02	0.6048	0.6184	0.6086	0.6278	0.6120	**0.6326**	0.6075	0.6031	0.6008	0.5326	0.5183	0.5455	0.5470
	k03	0.6317	0.6499	0.6366	0.6659	0.6414	**0.6766**	0.6324	0.6298	0.6265	0.6099	0.6025	0.5413	0.6097
	k04	0.6492	0.6707	0.6546	0.6905	0.6598	**0.7045**	0.6422	0.6501	0.6436	0.6147	0.6154	0.5562	0.6329
	k05	0.6689	0.6958	0.6751	0.7221	0.6812	**0.7409**	0.6808	0.6732	0.6625	0.6569	0.6512	0.5445	0.6535
	k06	0.6555	0.6833	0.6620	0.7120	0.6685	**0.7340**	0.6622	0.6651	0.6492	0.6163	0.5918	0.5095	0.5963
	k07	0.6719	0.6998	0.6782	0.7274	0.6845	**0.7482**	0.6709	0.6834	0.6657	0.6696	0.6396	0.5028	0.6443
T13-whole	k02	0.6111	0.6192	0.6139	0.6238	0.6162	**0.6254**	0.6005	0.6078	0.6079	0.5410	0.4974	0.5012	0.5186
	k03	0.6526	0.6616	0.6562	0.6692	0.6594	**0.6733**	0.6254	0.6548	0.6486	0.6088	0.5926	0.4770	0.6085
	k04	0.6687	0.6825	0.6728	0.6941	0.6765	**0.7008**	0.6250	0.6823	0.6641	0.6214	0.6119	0.4910	0.6241
	k05	0.7061	0.7237	0.7106	0.7387	0.7148	**0.7478**	0.6797	0.7209	0.7011	0.6613	0.6491	0.4478	0.6497
	k06	0.6872	0.7068	0.6923	0.7253	0.6971	**0.7379**	0.6502	0.7151	0.6818	0.6197	0.5913	0.4289	0.5919
	k07	0.7045	0.7232	0.7092	0.7402	0.7135	**0.7515**	0.6552	0.7330	0.6996	0.6708	0.6452	0.4062	0.6476
T08-partitioned	k02	0.5314	0.5390	0.5332	0.5456	0.5350	0.5500	**0.5508**	0.5317	0.5294	0.4919	0.4944	0.5024	0.4913
	k03	0.5466	0.5587	0.5497	0.5700	0.5526	0.5783	**0.5831**	0.5457	0.5436	0.5171	0.5292	0.5050	0.5321
	k04	0.5549	0.5690	0.5584	0.5830	0.5621	0.5935	**0.6037**	0.5553	0.5512	0.5153	0.4967	0.4992	0.5191
	k05	0.5564	0.5725	0.5604	0.5891	0.5645	0.6019	**0.6168**	0.5599	0.5523	0.5368	0.4804	0.4914	0.5118
	k06	0.5683	0.5863	0.5729	0.6064	0.5775	0.6226	**0.6552**	0.5692	0.5638	0.5287	0.4785	0.4782	0.4962
	k07	0.5672	0.5900	0.5737	0.6150	0.5797	0.6333	**0.6872**	0.5696	0.5615	0.5373	0.4774	0.4639	0.4776
T13-partitioned	k02	0.5842	0.5959	0.5862	0.6038	0.5879	**0.6078**	0.5998	0.5767	0.5820	0.5406	0.5052	0.4945	0.4847
	k03	0.6155	0.6299	0.6181	0.6406	0.6206	**0.6474**	0.6412	0.6015	0.6126	0.5728	0.5854	0.4611	0.5742
	k04	0.6372	0.6528	0.6402	0.6647	0.6430	**0.6722**	0.6706	0.6270	0.6340	0.5848	0.5757	0.4157	0.5838
	k05	0.6481	0.6641	0.6515	0.6773	0.6549	0.6862	**0.6929**	0.6508	0.6444	0.6079	0.5619	0.3521	0.6009
	k06	0.6616	0.6776	0.6653	0.6914	0.6691	0.7015	**0.7211**	0.6663	0.6579	0.6165	0.5573	0.3044	0.5840
	k07	0.6560	0.6728	0.6603	0.6884	0.6642	0.7006	**0.7306**	0.6412	0.6512	0.6209	0.5332	0.1963	0.5568
T26-partitioned	k02	0.3817	0.4008	0.3796	0.4084	0.3774	**0.4124**	0.3531	0.3928	0.3837	0.3731	0.3362	0.3506	0.3625
	k03	0.3863	0.4067	0.3839	0.4151	0.3815	**0.4191**	0.3522	0.4028	0.3886	0.3783	0.3512	0.3753	0.3680
	k04	0.3824	0.4072	0.3795	0.4179	0.3767	**0.4236**	0.3421	0.4029	0.3853	0.3791	0.3525	0.3688	0.3625
	k05	0.3832	0.4102	0.3796	0.4228	0.3761	**0.4295**	0.3396	0.4077	0.3866	0.3785	0.3602	0.3648	0.3729
	k06	0.3926	0.4232	0.3896	0.4366	0.3870	**0.4441**	0.3568	0.4207	0.3961	0.3781	0.3584	0.3466	0.3737
	k07	0.4534	0.4787	0.4521	0.4918	0.4507	**0.4980**	0.4171	0.4841	0.4561	0.4400	0.4302	0.3715	0.4239

and squared s-AWARE approaches achieve, in general, better performance than the basic closeness approach, since they emphasize more sharply the difference between good and bad assessors.

Figure 3 shows the interaction plot between k-tuple size and the different approaches. An interaction plot displays the levels of one factor on the X axis, k-tuple size in our case, and has a separate line for the means of each level of the other factor on the Y axis, approach effectiveness in terms of APC in our case. This plots allows us to understand whether the effect of one factor depends on the level of the other factor. Two parallel lines indicate that no interaction occurred, whereas nonparallel lines indicate an interaction between factors; the more nonparallel the lines are, the greater the strength of the interaction.

Figure 3a and 3c show the *Whole Assessors* case on `T08` and `T13`. We can observe how all the AWARE approaches, and especially the s-AWARE, better exploit small k-tuple sizes and grow more rapidly than the baselines as the k-tuple size increases. We can also note how the supervised `emGZ` approach struggles in effectively exploiting the higher k-tuple sizes.

In Figure 3b and 3d we consider the *Partitioned Assessors* case for `T08` and `T13`. Again, we can observe that AWARE approaches better interact with the k-size, even if in this context u-AWARE approaches dominate the scene, being this case easier for unsupervised approaches. Finally, Figure 3e highlights the good performance of s-AWARE on the `T26` track which is possibly the most realistic dataset..

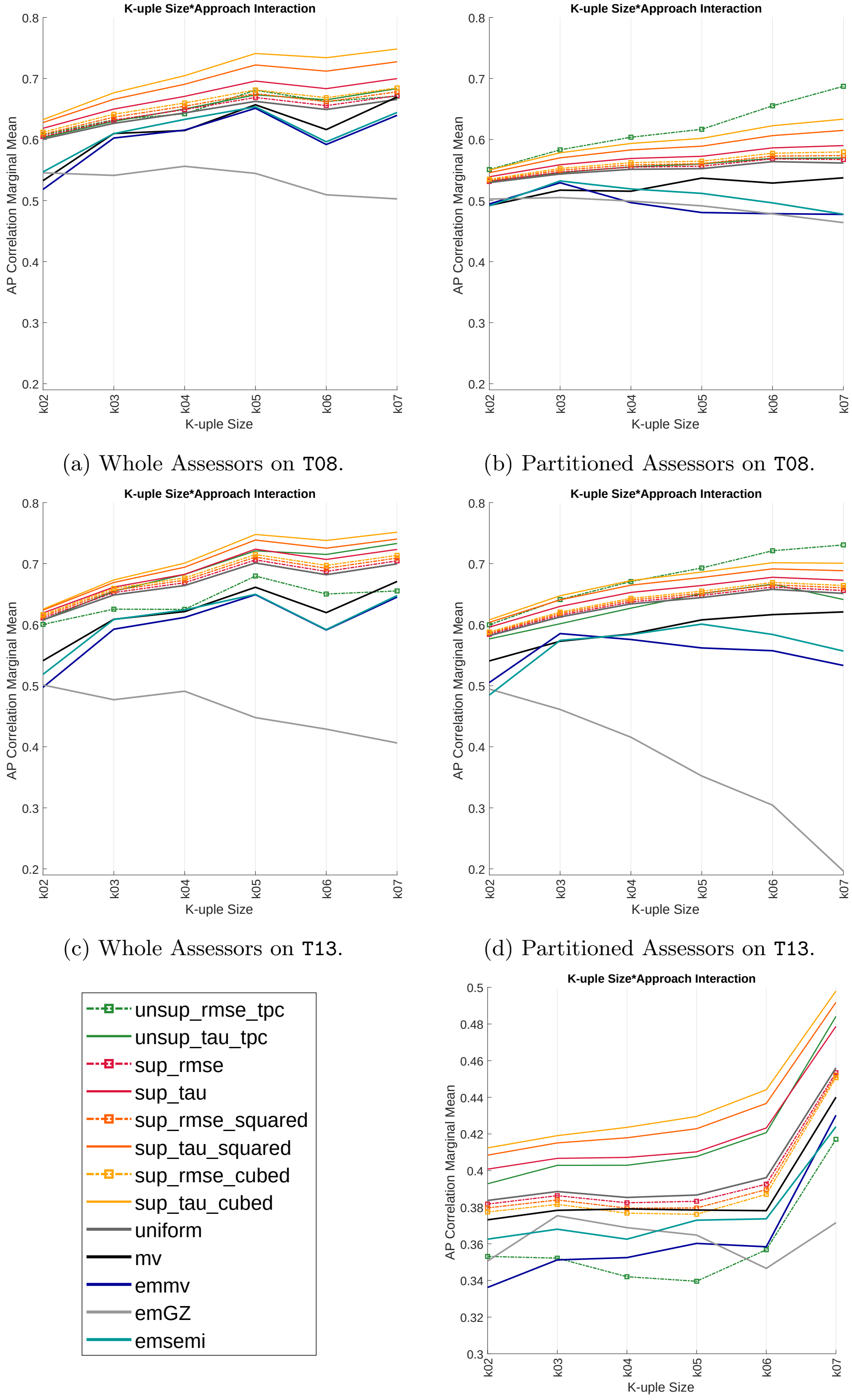
Overall, Figure 3 confirms and supports the previous observations about the differences between the various approaches when facing the *Whole Assessors* and *Partitioned Assessors* cases and highlight the strengths of the s-AWARE approaches.

5 Conclusions and Future Work

In this paper, we have faced the problem of effectively merging crowd-assessors and we have extended the AWARE approach to supervised techniques. We conducted an extensive experimental evaluation based on several TREC collections. We have evaluated approaches using few training data – just 30% for training and 70% for testing – since this is the most suitable, yet challenging, case for a real world scenario

We found that s-AWARE approaches outperform all the others in the *Whole Assessors* case and they are still quite robust also in a real scenario under the *Partitioned Assessors* case. Moreover, supervised and unsupervised AWARE approaches perform consistently better than the analyzed state-of-the-art approaches and they are especially effective at small k-tuple sizes, i.e. fewer crowd-assessors, making them more attractive for real world settings.

Future work will investigate how to extend AWARE approaches to better deal with sub-assessors, i.e. the *Partitioned Assessors* case, by allowing for multiple a_k scores for a topic, each one corresponding to a different sub-assessor.



(a) Whole Assessors on T08.

(b) Partitioned Assessors on T08.

(c) Whole Assessors on T13.

(d) Partitioned Assessors on T13.

(e) Partitioned Assessors on T26.

Fig. 3: Interaction plots between approach and k-tuple size.

References

[1] Allan, J., Harman, D.K., Kanoulas, E., Li, D., Van Gysel, C., Voorhees, E.M.: TREC 2017 Common Core Track Overview. In: Voorhees, E.M., Ellis, A. (eds.) The Twenty-Sixth Text REtrieval Conference Proceedings (TREC 2017), National Institute of Standards and Technology (NIST), Special Publication 500-324, Washington, USA (2018)

[2] Alonso, O.: The Practice of Crowdsourcing. Morgan & Claypool Publishers (2019)

[3] Alonso, O.: The Practice of Crowdsourcing. Morgan & Claypool Publishers, USA (May 2019)

[4] Alonso, O., Mizzaro, S.: Using crowdsourcing for trec relevance assessment. Inf. Process. Manage. **48**(6), 1053–1066 (Nov 2012), ISSN 0306-4573, https://doi.org/10.1016/j.ipm.2012.01.004, URL `http://dx.doi.org/10.1016/j.ipm.2012.01.004`

[5] Clough, P., Sanderson, M., Tang, J., Gollins, T., Warner, A.: Examining the limits of crowdsourcing for relevance assessment. IEEE Internet Computing **17**(4), 32–38 (jul 2013), https://doi.org/10.1109/mic.2012.95, URL `https://doi.org/10.1109%2Fmic.2012.95`

[6] Ferrante, M., Ferro, N., Maistro, M.: AWARE: Exploiting Evaluation Measures to Combine Multiple Assessors. ACM Transactions on Information Systems **36**(2), 1–38 (aug 2017), https://doi.org/10.1145/3110217, URL `https://doi.org/10.1145%2F3110217`

[7] Georgescu, M., Zhu, X.: Aggregation of crowdsourced labels based on worker history. In: Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14), WIMS '14, Association for Computing Machinery, New York, NY, USA (2014), ISBN 9781450325387, https://doi.org/10.1145/2611040.2611074, URL `https://doi.org/10.1145/2611040.2611074`

[8] Hosseini, M., Cox, I.J., Milić-Frayling, N., Kazai, G., Vinay, V.: On aggregating labels from multiple crowd workers to infer relevance of documents. In: Proceedings of the 34th European Conference on Advances in Information Retrieval, pp. 182–194, ECIR'12 (2012)

[9] Inel, O., Haralabopoulos, G., Li, D., Van Gysel, C., Szlávik, Z., Simperl, E., Kanoulas, E., Aroyo, L.: Studying Topical Relevance with Evidence-based Crowdsourcing. In: Cuzzocrea, A., Allan, J., Paton, N.W., Srivastava, D., Agrawal, R., Broder, A., Zaki, M.J., Candan, S., Labrinidis, A., Schuster, A., Wang, H. (eds.) Proc. 27th International Conference on Information and Knowledge Management (CIKM 2018), pp. 1253–1262, ACM Press, New York, USA (2018)

[10] Nellapati, R., Peerreddy, S., Singhal, P.: Skierarchy: Extending the power of crowdsourcing using a hierarchy of domain experts, crowd and machine learning. In: Proceedings of the TREC 2012 crowdsourcing track, pp. 1–11 (2012)

[11] Sanderson, M.: Test Collection Based Evaluation of Information Retrieval Systems. Foundations and Trends in Information Retrieval (FnTIR) **4**(4), 247–375 (2010)

[12] Smucker, M.D., Kazai, G., Lease, M.: Overview of the TREC 2012 Crowdsourcing Track. In: Voorhees, E.M., Buckland, L.P. (eds.) The Twenty-First Text REtrieval Conference Proceedings (TREC 2012), National Institute of Standards and Technology (NIST), Special Publication 500-298, Washington, USA (2013)

[13] Tang, W., Lease, M.: Semi-supervised consensus labeling for crowdsourcing. In: Proceedings of the SIGIR 2011 workshop on crowdsourcing for information retrieval (CIR), pp. 36–41, Association for Computing Machinery, New York, NY, USA (2011), ISBN 9781450325387

[14] Tao, D., Cheng, J., Yu, Z., Yue, K., Wang, L.: Domain-weighted majority voting for crowdsourcing. IEEE Transactions on Neural Networks and Learning Systems **30**(1), 163–174 (jan 2019), https://doi.org/10.1109/tnnls.2018.2836969, URL `https://doi.org/10.1109%2Ftnnls.2018.2836969`

[15] Tian, T., Zhu, J., Qiaoben, Y.: Max-margin majority voting for learning from crowds. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(10), 2480–2494 (oct 2019), https://doi.org/10.1109/tpami.2018.2860987, URL `https://doi.org/10.1109%2Ftpami.2018.2860987`

[16] Voorhees, E.M.: Overview of the TREC 2004 Robust Track. In: Voorhees, E.M., Buckland, L.P. (eds.) The Thirteenth Text REtrieval Conference Proceedings (TREC 2004), National Institute of Standards and Technology (NIST), Special Publication 500-261, Washington, USA (2004)

[17] Voorhees, E.M., Harman, D.K.: Overview of the Eigth Text REtrieval Conference (TREC-8). In: Voorhees, E.M., Harman, D.K. (eds.) The Eighth Text REtrieval Conference (TREC-8), pp. 1–24, National Institute of Standards and Technology (NIST), Special Publication 500-246, Washington, USA (1999)

[18] Whiting, S., Perez, J., Zuccon, G., Leelanupab, T., Jose, J.: University of glasgow (qirdcsuog) at trec crowdsourcing 2001: Turkrank – network based worker ranking in crowdsourcing. In: Proceedings of The Twentieth Text REtrieval Conference, TREC 2011, Gaithersburg, Maryland, USA, November 15-18, 2011, pp. 1–7 (jan 2011)

[19] Yilmaz, E., Aslam, J.A., Robertson, S.E.: A New Rank Correlation Coefficient for Information Retrieval. In: Chua, T.S., Leong, M.K., Oard, D.W., Sebastiani, F. (eds.) Proc. 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008), pp. 587–594, ACM Press, New York, USA (2008)