Predicting Opinions Across Multiple Issues in Large Scale Cyber Argumentation Using Collaborative Filtering and Viewpoint Correlation

Md Mahfuzer Rahman
University of Arkansas
Fayetteville, AR, USA
email: mmr014@uark.edu

Joseph W Sirrianni
University of Arkansas
Fayetteville, AR, USA
email: jwsirria@uark.edu

Xiaoqing (Frank) Liu
University of Arkansas
Fayetteville, AR, USA
email: frankliu@uark.edu

Douglas Adams
University of Arkansas
Fayetteville, AR, USA
email: djadams@uark.edu

***Abstract*—One challenging problem in large-scale cyber argumentation is that discussions are often incomplete as some ideas only get addressed by a fraction of the users. Typically, users engage only with some ideas but not all of them, making it difficult to assess collective intelligence. To resolve this problem, we developed an innovative method of predicting a user's opinion on ideas that they have not discussed using the opinions from related ideas with intelligent argumentation and collaborative filtering. Our method considers the similarity of users and the correlation of different ideas across issues to make predictions. Compared to other existing opinion prediction methods, experimental results on an empirical dataset show that our method is 21.7% more accurate. Two major innovative contributions are made in this research: 1) We developed a novel approach to predict a participant's opinion on a non-participated idea using similar users' opinions from related ideas with an excellent accuracy in cyber argumentation; 2) This is the first research to enable multi issue opinion prediction with partial agreement on an idea. This is encouraging from several perspectives. This prediction model will help to assess collective intelligence from cyber argumentation more accurately by providing additional data both in individual and collective level. In addition, it may speed up a cyber argumentation analysis process by reducing the amount of participation required.**

Keywords-opinion prediction; incomplete ongoing discussion; collaborative filtering; cyber argumentation; collective intelligence.

I. INTRODUCTION

In large-scale cyber argumentation platforms, participants express their opinions, engage with one another and respond to feedback and criticism from others in discussing important issues online. Cyber argumentation platforms implement argumentation models to enforce an explicit discussion structure, such as Dung abstract frameworks [1], Issue-Based Information Systems (IBIS) [2], and Toulmin's model of argumentation [3]. These structures allow argumentation analysis tools to effectively analyze the discussions. Argumentation analysis tools can capture the collective intelligence of the participants and reveal hidden insights from the underlying discussions. In this research domain, these tools have demonstrated the ability to evaluate and reveal hidden phenomena, such as identifying group-think [4], polarization [5], assessing argument validity [1], etc.

However, such analysis requires that the issues have been thoroughly discussed and participant's opinions are clearly expressed and understood. Participants typically focus only on few ideas and leave others unacknowledged and under-discussed. This generates a limited dataset to work with resulting in an incomplete analysis of issues in the discussion. This also hampers the individual and collective intelligence retrieval process and opinion analysis from the underlying discussion. Particularly a limited dataset with missing values affects the clustering or user grouping algorithms and the resulting user groups introduce error and bias in different social phenomena analysis [6].

One solution to this problem would be to predict a participant's opinion with high accuracy on an idea that they have not explicitly expressed. With reasonably accurate prediction of missing information, we can analyze the individual and collective opinion of users effectively even if they did not participate in some of the discussion. Collective intelligence can also be assessed more accurately when discussions are incomplete. Predicted values can also fill the missing information for clustering algorithms and the derived group related analytical models.

In this paper, we present a method of predicting participant's opinions on different ideas that they have not explicitly engaged with. We use our argumentation platform, the Intelligent Cyber Argumentation System (ICAS), to collect user opinion on issues and predict the missing opinions. In our system, discussions take on a tree structure. Issues are the root of the conversation. Under an issue, there are a finite set of different positions that address the issue. We use a collaborative filtering model based on viewpoint correlation between positions and user opinion similarity to predict user's missing opinion on a position.

We compared our method Cosine Similarity with Correlation based Collaborative Filtering (CSCCF), with other opinion prediction methods based on popular predictive techniques on an empirical dataset collected with our argumentation platform, ICAS. Our dataset contains over ten thousand arguments on four issues and sixteen associated positions from more than three hundred participants. The experimental results show that our model has good accuracy and is 21.7% more accurate on average than other benchmarking methods.

In this paper, we make the following contribution:

- We propose a model (CSCCF) for predicting user opinion on positions using collaborative filtering based on viewpoint correlation between positions and user opinion similarity.
- We compare our model with other popular predictive techniques on an empirical dataset and show that our method is more accurate.

 ISBN: 978-1-61208-757-3

- We demonstrate how our method is capable of predicting several different positions at once without significantly compromising accuracy.

The rest of the paper is structured in the following way. In Section 2, we discuss previous research works which are related in different aspects/ways with the work presented in this paper. In section 3, we give a brief description about our argumentation platform ICAS and how we derive user's opinion in different issues. Section 4 describes the CSCCF opinion prediction model to predict missing opinion values. In Section 5, we talk about the empirical study to collect dataset, and different experiments to evaluate our CSCCF model. The remaining sections contains the Discussion, Conclusion and Reference for this work.

II. RELATED WORK

This section describes previous research works which are related in four different aspects with our work presented in this paper.

A. *Opinion Analysis on Argumentation Platform*

Many researchers have worked on analyzing user opinion in cyber argumentation system, such as opinion space [7] and Considerit [8] etc. Their main objective was analyzing how users engage with different opinionated people or ideas and how it affects their overall opinion. These platforms mostly focused on analyzing collective user opinion from user participation data only. None of these platforms have attempted to predict user opinion on non-participated issues.

B. *Opinion Prediction on Social Media*

Social media data is often used by many researchers to work on collective user stance/opinion prediction. Political discussions on twitter have been used to classify user political stance [9]. Social media data was also used to predict user reaction on certain events, such as the 2015 Paris Terror Attack [10] or classify people's stance on important issues [11]. These works mostly looked at predicting opinion on a single issue using the related textual content on that issue only, they are not using the user opinion in related issues to infer opinion in another issue like our method presented in this paper.

C. *Multi-Issue Opinion Prediction*

Little work has been done on an individual's opinion prediction across multiple issues. [12] used Probabilistic Matrix Factorization (PMF) to fill out a user-aspect opinion matrix (aspects are analogous with issues) as an intermediate step of a larger process to predict the polarity of interaction between users. However, since this was an intermediate step, the authors did not evaluate the success of the prediction step. [13] used traditional collaborative filtering methods to predict user's opinion on important political topics. In a follow-up paper [14], they used topic distribution from user arguments, user interaction and profile data to infer a user's stance on an issue. In their system, each issue only had two positions and users can only agree or disagree with a position. Whereas in our system each issue can have multiple positions and user can agree or disagree with a level of agreement from -1.0 to +1.0.

D. *Different Variation of Collaborative Filtering*

One of the major differences between different memory based collaborative filtering (CF) algorithms is how they calculate similarity between users/items to predict missing values from the most similar users/items. One popular approach measures the correlation between two users/items and use it as a similarity measurement between them [15], such as Pearson Correlation, Kendall's τ correlation. Cosine similarity of two user/item vectors is also used to measure similarity among them [15]. To our knowledge there is no similarity method that uses correlation values of items as weight in cosine similarity measurement like our method.

Some CF approaches measure the correlation values between different data domains. Collective Link Prediction, and Multi-domain Collaborative Filtering [16] are some of the models which exploit domain correlation via different learning based methods. Collective or Relational Matrix factorization [17] models use correlation between multiple relations for relational learning when an entity/user participates in multiple relations. Cross domain CF model uses this approach via coordinate system transfer method [16]. However, these models are computationally expensive and used to figure out correlations in between different data domains or multiple relations. Whereas, our model exploits the correlation within one data domain or in a single relation between user and item in a computationally inexpensive way.

III. ICAS SYSTEM

We use our intelligent cyber argumentation platform ICAS to derive viewpoint vectors for each participant, which are later used for opinion prediction. ICAS is a cyber-argumentation platform that is capable of automatically determining the opinion of participants towards different positions in the discussion. ICAS is the enhanced version of the online argumentation system developed in prior work [18].

A. *ICAS Architecture*

In the ICAS architecture, discussions take on a tree structure, with issues at the root of the tree, positions solving/addressing the root issue on the first level of the tree, and the arguments made for or against the positions or other arguments in the position as the remaining nodes in the tree. Participants contribute to the discussion by making arguments. Arguments are statements of agreement (for or against) and rationale relating to their parent node. Arguments can be made to support/attack positions or refute/agree with other arguments. When writing an argument, participants fill out two fields. First is the argument text, where they give their rationale for the argument. The second is the level of agreement. Here, users choose their level of agreement on a weighted scale from -1.0 to +1.0 at 0.2 length intervals. The sign of the agreement level indicates whether the user is agreeing (positive) or disagreeing (negative) with the parent node. The magnitude of the agreement level indicates the intensity of the agreement, where a lower magnitude is closer to indifference and a greater magnitude is closer to complete agreement/disagreement. For example, an agreement level of +0.8 would represent a very high level of support, while an

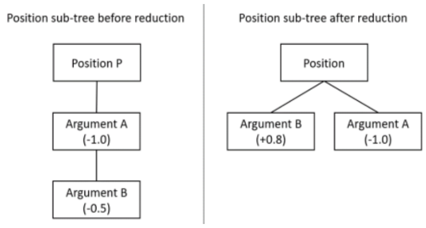

Figure 1. Example of an argument reduction. Argument B is reduced from the second level of the tree to the first level.

agreement level of -0.4 would represent a moderate level of disagreement.

B. *Deriving Viewpoint Vectors using ICAS*

A viewpoint vector is a vector where each element represents a user's opinion toward a position being discussed. We average the agreement values of the arguments a user posted under a position to determine a user's opinion toward the position. A user can post arguments supporting or attacking other user's arguments at different levels of argument tree. The associated agreement values state user's agreement with the parent argument, not with the root position directly. We used argument reduction method [19] to connect all these arguments to the root position. This method uses artificial intelligence, fuzzy logic, linguistic heuristic rules and other techniques to reduce an argument from any level of argument tree to the first level considering the support/ attack relationships with updated agreement value. The updated value represents the argument's agreement value directly towards the root position. Fig. 1 visualizes this reduction. For a more in-depth explanation of the fuzzy logic engine and argument reduction method, refer to [19, 20]. Argument reduction method is not 100% accurate, instead this is an estimation of user's opinion towards the root position. Several case studies have shown that this method achieves reasonable accuracy [19][20].

IV. OPINION PREDICTION MODEL

The section describes the CSCCF model for missing opinion value prediction. It is divided into three sub sections which describes required data for CSCCF model, steps and algorithms for CSCCF, and time complexity of CSCCF model.

A. *Data Required for Prediction*

To predict a missing opinion value at a position we need the following information: 1) the viewpoint vectors of all users in the training data, 2) opinion correlations of different positions with the target position t, and 3) the target user's viewpoint vector.

A viewpoint vector represents the opinions or agreement values of a particular user for all positions in the system. At training time, our model calculates the viewpoint vectors for every user. If there are n different positions on various issues in the system, we can represent a user's viewpoint vector in the following format:

$U_i = [R_1^i, R_2^i, R_3^i, R_4^i \dots\dots\dots R_n^i]$; here U_i is the viewpoint vector for user i and R_p^i is the opinion value of the user i at position p. If user i did not participate in position p discussion, then R_p^i will be represented as invalid or missing value.

The correlation value between two positions indicates how much participant's opinions are associated in these two positions. A strong correlation value indicates if a user agreed in one position, whether the user agreed or disagreed in another position and vice versa. The correlation vector of a position can be formalized in the following format:

$C_p = [C_{p1}, C_{p2}, C_{p3}, \dots, C_{pn}]$; here C_p is the correlation vector of position p and C_{pq} is the correlation between position p and position q, C_{pp} would represent the correlation value between the same position p, which is 1. Although this value will not be used in predicting position p, only the related positions with position p will be used. We calculated the correlation values between positions using the Pearson Correlation Coefficient from the training data and only considered the correlation values with high confidence (Two tailed p-values above 0.05 are discarded).

The target user's viewpoint vector can be represented in the following format:

$U_x = [R_1^x, R_2^x, R_3^x, R_4^x \dots, R_{t-1}^x, ?, R_{t+1}^x \dots\dots R_n^x]$; Here, R_t^x, the value at position t is missing, we will predict this value.

B. *Opinion Prediction using Cosine Similarity and Position Correlation*

We want to predict the opinion value of user x at position t, the R_t^x value in U_x. This process has two steps. First, we need to identify the most similar users to user x with respect to position t from our training data. Second, we need to aggregate their opinion values at position t to use it as predicted value.

To identify the most similar users with respect to position t, we filter out the users who have a missing value at position t in their viewpoint vector. The remaining users are placed into user x's candidate set. Then the similarity between target user x and every user in the candidate set is calculated.

To calculate similarity between two users x and y, we first remove any elements from the vectors at which either vector has a missing value. Given, U_x and U_y are the viewpoint vectors of user x and users y. U_x has a missing value at position t, so we remove R_t^x and R_t^y from the vectors.

$U_x = [R_1^x, R_2^x \dots R_{t-1}^x, R_{t+1}^x \dots\dots\dots R_n^x]$

$U_y = [R_1^y, R_2^y \dots R_{t-1}^y, R_{t+1}^y \dots\dots\dots R_n^y]$

Next, the viewpoint vectors are updated using the values from target position's correlation vector, C_t. Each value in the viewpoint vector is multiplied by its corresponding position correlation value with target position t. The updated viewpoint vectors are represented as $U_x^\wedge$ and $U_y^\wedge$:

$U_x^\wedge = [C_{t,1}R_1^x, C_{t,2}R_2^x, \dots, C_{t,t-1}R_{t-1}^x, C_{t,t+1}R_{t+1}^x, \dots, C_{t,n}R_n^x]$

$U_y^\wedge = [C_{t,1}R_1^y, C_{t,2}R_2^y, \dots, C_{t,t-1}R_{t-1}^y, C_{t,t+1}R_{t+1}^y, \dots, C_{t,n}R_n^y]$; here, Opinion value at position i is multiplied by $C_{ti;}$ the correlation value between position i and t.

Then, we calculate the cosine similarity between the updated viewpoint vector $U_x^\wedge$ and $U_y^\wedge$ to determine how similar user x and y are with respect to position t using (1).

 ISBN: 978-1-61208-757-3

The similarity value lies in between [-1,1], where -1 represents complete difference, 0 represents no correlation, and 1 represents complete similarity.

Similarity (user x, user y) = Cosine Similarity ($U_x^\wedge$, $U_y^\wedge$)

$$= \frac{\sum_{i=1,i\neq t}^{n} C_{ti}^2 R_i^x R_i^y}{\sqrt{\sum_{i=1,i\neq t}^{n} C_{ti}^2 (R_i^X)^2} + \sqrt{\sum_{i=1,i\neq t}^{n} C_{ti}^2 (R_i^y)^2}} \quad (1)$$

Using the above method, we calculate the similarity between target user x and every user in x's candidate set. Then, we rank all the users based on their similarity value with target user x and select the top k neighbors, where k is a constant model parameter. We experimented with different values for k (3, 5, 10 etc.), we got the best result when k was set at 5 on the dataset we validated this model. The model then averages the opinion values of top k neighbors at position t weighted by the similarity value to predict the value of R_t^x as shown in (2).

$$\text{Predicted value of } R_t^x = \frac{\sum_{m=1}^{k} Similarity(x,m) * R_t^m}{\sum_{m=1}^{k} Similarity(x,m)} \quad (2)$$

Our method finds the most similar users to the target user with respect to the position we are predicting. Multiplying the opinion values with the associated test position correlation values weights the opinion values as per their importance to determine the test position. It also filters out the uncorrelated opinion values in similarity calculation.

C. *Time Complexity of CSCCF Model*

Let, number of users = n and number of positions = m, we will measure the time complexity to predict a missing opinion value for one test user. We calculate the correlation values between the positions from the training data only one time and use it to predict the missing opinions for all test users. To make one single prediction, first we calculate the cosine similarity between updated viewpoint vectors n times, one for each user. Then, we sort the similarity values from n users and make prediction from top k neighbors. The time complexity of these two steps are O(n*m) and the time complexity of sorting n numbers respectively. In our case, the time complexity of sorting n number was O(nlogn) as we used heap-based priority queue. So, the overall time complexity of our algorithm is O(n*m) + O(nlogn).

V. EXPERIMENTS

This section describes the empirical study, dataset collection process and experimental setup to evaluate our CSCCF model.

A. *Empirical Data Description*

We conducted an empirical study in spring of 2018 on a group of 344 undergraduate students in an entry level sociology class. The students were asked to discuss four issues, each with 4 different positions over the course of five weeks. The resulting discussion had over 10000 arguments, from 309 users. 90 out of 309 users had complete participation. On average 69 users (22.33%) had missing opinion values in the positions. We received Institutional Review Board (IRB) approval from the university to conduct this empirical study and use the anonymized data for research purposes. Table 1 describes the dataset with issues and positions.

B. *Methods to Test Against*

We tested our model (CSCCF) against following different popular predictive techniques to compare accuracy. The only difference between CSCCF and other CF based models is the way similarity between two users is measured.

1) *Cosine Similarity based CF (CSCF) :* This CF model used the Cosine similarity between the original viewpoint vectors U_x and U_y, to calculate similarity between user x and y using (3):

$$\text{Cosine Similarity } (U_x, U_y) = \frac{\sum_{i=1,i\neq t}^{n} R_i^x R_i^y}{\sqrt{\sum_{i=1,i\neq t}^{n} (R_i^X)^2} + \sqrt{\sum_{i=1,i\neq t}^{n} (R_i^y)^2}} \quad (3)$$

2) *Neural Net :* We implemented a neural net that uses hybrid latent variables as described in [21] to learn individual

TABLE I. DATA DESCRIPTION WITH ISSUES AND POSITIONS

Issue Name	Position No	Position Text
Guns on Campus: Should students with a concealed carry permit be allowed to carry guns on campus?	0	No, college campuses should not allow students to carry firearms under any circumstances.
	1	No, but those who receive special permission from the university should be allowed to concealed carry.
	2	Yes, but students should have to undergo additional training.
	3	Yes, and there should be no additional test. A concealed carry permit is enough to carry on campus.
Religion and Medicine: Should parents who believe in healing through prayer be allowed to forgo medical treatment for their child?	4	Yes, religious freedom should be respected.
	5	Yes, but only in cases where the child's life is not in immediate danger.
	6	No, but may deny preventative treatments like vaccines.
	7	No, the child's medical safety should come first.
Same Sex Couples and Adoption: Should same sex married couples be allowed to adopt children?	8	No, same sex couples should not be allowed to legally adopt children.
	9	No, but adoption should be allowed for blood relatives of the couple, such as nieces/nephews.
	10	Yes, but same sex couples should have special vetting to ensure that they can provide as much as a heterosexual couple.
	11	Yes, same sex couples should be treated the same as heterosexual couples and be allowed to adopt via the standard process.
Government and Healthcare: Should individuals be required by the government to have health insurance?	12	No, the government should not require health insurance.
	13	No, but the government should provide help paying for health insurance.
	14	Yes, the government should require health insurance and help pay for it, but uninsured individuals will have to pay a fine.
	15	Yes, the government should require health insurance and guarantee health coverage for everyone.

ISBN: 978-1-61208-757-3

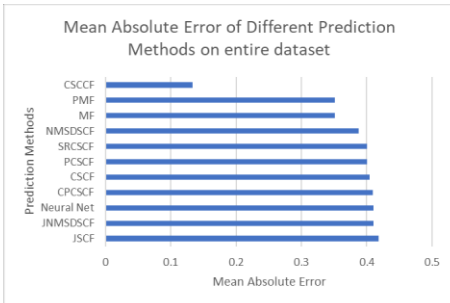

Figure 2. Mean Absolute Error of different Models on entire dataset

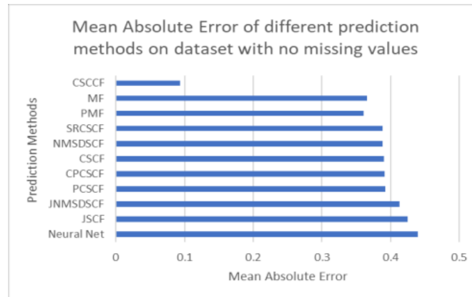

Figure 3. Mean Absolute Error of different Models with no missing values

information about both the users and positions. The neural net learns model weights and latent input variables during training. Various input latent vector sizes were tried, latent vectors with length 2 for both users and positions did best. The topology of the neural net is: linear layer(4, 6) => Tanh layer(6,6) => linear layer(6,1) => Tanh layer(1, 1). The first argument for the layer is the input size, the second is the output size. The neural net used stochastic gradient descent and optimized for sum squared error (SSE).

3) Matrix Factorization (MF) : We implemented Regularized Incremental Simultaneous MF as described in [22] which decomposes the user-position matrix (|U| * |D|) into two matrices (|U| * |K| and |D| * |K|) to discover K latent features. In oreder to avoid overfitting, this method applies regularization by penalizing the magnitude of vectors. It was optimized for SSE. We tried different sizes for latent factor K, the best result was found when K was 5.

4) Probabilistic Matrix Factorization (PMF) : We implemented PMF as described in [23]. The latent matrices are drawn from a gaussian distribution, determined by the means and variances of each row in the original user-position matrix and was optimized for SSE. Different latent factor sizes were tried and 5 did best for PMF.

5) Spearman Rank Correlation Similarity based Collaborative Filtering (SRCSCF): We ranked the original viewpoint vector (U_x and U_y) and measured the similarity between user x and y using (4):

$$Sim(user\ x, user\ y) = 1 - \frac{6\sum_{h=0}^{n} d_h^2}{n(n^2-1)} \tag{4}$$

Here, d_h is the difference in the ranks for item h by the user x and y, n is the number of co-rated items.

6) Pearson Correlation Similarity based Collaborative Filtering (PCSCF) : Pearson correlation coefficient value of U_x and U_y is used to measure similarity between users.

7) Constrained Pearson Correlation Similarity based Collaborative Filtering (CPCSCF) : This method uses midpoint instead of mean value from U_x and U_y in Pearson correlation to measure similarity between users.

8) Jaccard Similarity based Collaborative Filtering (JSCF) : We have rounded the opinion agreement values in U_x and U_y upto two decimal points and measured the Jaccard coefficient as similarity using (5):

$$Sim(user\ x, user\ y\ ) = \frac{|U_x' \cap U_y'|}{|U_x' \cup U_y'|} \tag{5}$$

9) Normalized Mean Squared Difference Similarity based Collaborative Filtering (NMSDSCF): It uses the normalized mean squared difference (NMSD) of rating vectors as difference between users to calculate similarity.

10) Jaccard and Mean Squared Difference Similarity based Collaborative Filtering (JNMSDSCF) : This method multiplies similarity value from JSCF and NMSDCF to calculate similarity between users.

C. Results

We tested our model CSCCF along with other comparison models and measured the Mean Absolute Error (MAE) from the predicted and actual opinion value for the following experiments. We performed a cross validation with 5 fold and 2 repetitions and the data was separated as 80% training and 20% testing in each iteration.

1) Accuracy on entire dataset: We measured the MAE for each position separately and then averaged the results. Fig. 2 summarizes the result of this experiment. On average our model achieved a MAE value of 0.133. The second most accurate model, PMF achieved a MAE value of 0.350 and the other models were all in between 0.351 to 0.42. This shows that our model is a distinct improvement over other models. As most of the users did not participate in all positions, this dataset contains lots of missing information which is hampering other models. Our model handled this sparsity problem incorporating global correlation values from training data and used them as weight to prioritize the limited available opinion values in the similarity calculation.

2) Accuracy on dataset with no missing values: We also tested how the accuracy of different models would change if we only consider the data with no missing values. This dataset is much smaller as only 90 out of the 309 students

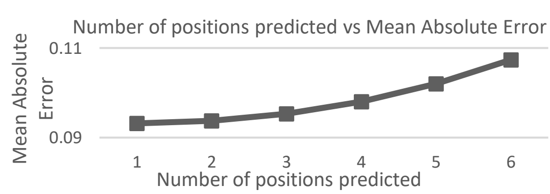

Figure 3. Number of positions prediction vs Mean Absolute Error

participated in all the position discussions. Fig. 3 summarizes the results of this experiment. MAE value of all the models tended to decrease for this dataset. On average our model's MAE decreased to 0.093 and for the second-best model, PMF it was 0.365. These results show that our model outperformed other models even on a complete dataset with no missing values. We think smaller dataset is the main reason of higher MAE values from neural net, and matrix factorization based models. And Prioritizing opinion values in similarity calculation is helping us to achieve lower MAE value than other CF based models.

3) Predicting multiple positions: We tested the accuracy of our model if we predict 2 to 6 positions simultaneously. At each number of predictions we considered all possible combination of position indices. As example, when we predict 2 positions at once, we tested with all 120 position combination of two position indices as testing positions and averaged the MAE values. Fig. 3 shows the result from this experiment on the dataset with no missing values. The MAE increases when more positions are being predicted at once. After 3 positions, the MAE increases at a faster rate but remains relatively low. The main cause of higher MAE is that we might be predicting correlated positions simultaneously. For example, if we are predicting correlated positions 1 and 3, position 3's value will not be used in postion 1's similarity calculation and vice versa.

VI. Discussion

We think our model is working because people's opinions are correlated across different issues due to their similarity in terms of their values in the sense of Schwartz theory of basic human values [24]. Their political leanings, such as conservative, lean conservative, lean liberal, liberal etc. and their position on religion are few of the issues deriving from their values. Generally people choose a certain perspective on social issues based on their political leanings which our model captures using the correlation values between positions.

The improvement over CF models notably the CSCF shows the importance of using viewpoint correlations in opinion prediction. Each opinion value had the same priority in similarity calculation in these models whereas in our model opinion values were weighted according to its correlation with the test position. The improvement over the neural net, MF, and PMF methods is likely because of the limited data size. The latent features for the users and positions were probably underdeveloped and contained little meaningful information. If each user had more data points, then these models might have done better. There is no straightforward way to filter out uncorrelated positions in these models. Neural Net automatically figures out which features are irrelevant, but the lack of data is preventing it from doing it. In CF or MF, missing values are predicted on an initial user-item matrix, there is no common way to filter out different item set for different item predictions.

If there is a strong correlation between the ratings of different data items from the overall users, we think our CSCCF model will generate good prediction results. Also, it might help to deal with the cold start and sparsity problem especially when a user has provided very few opinions on related issues. In order to achieve high accuracy by our CSCCF model, the data items need to be correlated by some degree. If there is no correlation among data items, then our model would not work as it will filter out all uncorrelated data items.

VII. Conclusion

In this paper, we developed an innovative opinion prediction method in large scale cyber argumentation on multiple issues. Our method predicts how much a user would agree with a position on an issue based on the opinions of similar users on related issues. Our model achieved an excellent accuracy with a MAE value of 0.133 using collaborative filtering and correlations between positions across issues. We assessed the impact of number of positions predicted, and degree of correlation on the opinion prediction accuracy in multi-issue cyber argumentation. The method uses correlation to achieve high accuracy, thus it cannot work for discussions that are not related. Relevancy between discussions should be kept in mind when using this model. Using this model participants' opinions on related issues can be assessed even when they haven't explicitly discussed them. Additionally, discussions with a small number of participants can be analyzed more representatively. The predicted values can be used to impute missing values for different clustering algorithms for different opinionated group related analytical models. It can also be used to assess collective thoughts even when cyber argumentation on multiple issues is incomplete.

References

[1] P. M. Dung, "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games," Artificial Intelligence, vol. 77, no. 2, pp. 321–357, Sep. 1995.

[2] W. Kunz and H. W. J. Rittel, "Issues as elements of information systems," Inst. Urban and Regional Devt., Univ. Calif. at Berkeley, 1970.

[3] S. E. Toulmin. 1958. The Uses of Argument. Cambridge, UK: University Press, 1958.

[4] M. Klein, "The CATALYST Deliberation Analytics Server," Social Science Research Network, Rochester, NY, SSRN Scholarly Paper, Nov. 2015.

[5] J. Sirrianni, X. Liu, and D. Adams, "Quantitative Modeling of Polarization in Online Intelligent Argumentation and Deliberation for Capturing Collective Intelligence," 2018 IEEE International Conference on Cognitive Computing (ICCC), pp. 57–64, 2018.

[6] S. Zhang, J. Zhang, X. Zhu, Y. Qin, and C. Zhang, "Missing Value Imputation Based on Data Clustering," in Transactions on Computational Science I, M. L. Gavrilova and C. J. K. Tan, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 128–138.

[7] S. Faridani, E. Bitton, K. Ryokai, and K. Goldberg, "Opinion Space: A Scalable Tool for Browsing Online Comments," in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, 2010, pp. 1175–1184.

[8] T. Kriplean, J. Morgan, D. Freelon, A. Borning, and L. Bennett, "Supporting Reflective Public Thought with Considerit," in

Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work, New York, NY, USA, 2012, pp. 265–274.

[9] M. Boireau, "Determining Political Stances from Twitter Timelines: The Belgian Parliament Case," in Proceedings of the 2014 Conference on Electronic Governance and Open Society: Challenges in Eurasia, New York, NY, USA, 2014, pp. 145–151.

[10] W. Magdy, K. Darwish, N. Abokhodair, A. Rahimi, and T. Baldwin, "#ISISisNotIslam or #DeportAllMuslims?: Predicting Unspoken Views," in Proceedings of the 8th ACM Conference on Web Science, New York, NY, USA, 2016, pp. 95–106.

[11] O. Fraisier, G. Cabanac, Y. Pitarch, R. Besançon, and M. Boughanem, "Stance Classification Through Proximity-based Community Detection," in Proceedings of the 29th on Hypertext and Social Media, New York, NY, USA, 2018, pp. 220–228.

[12] M. QIU, "Mining user viewpoints in online discussions," Dissertations and Theses Collection (Open Access), pp. 1–119, Jan. 2015.

[13] S. Gottipati, M. Qiu, L. Yang, F. Zhu, and J. Jiang, "Predicting User's Political Party Using Ideological Stances," in Social Informatics, 2013, pp. 177–191.

[14] M. Qiu, Y. Sim, N. A. Smith, and J. Jiang, "Modeling User Arguments, Interactions, and Attributes for Stance Prediction in Online Debate Forums," in Proceedings of the 2015 SIAM International Conference on Data Mining, 2015, pp. 855–863.

[15] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," Adv. in Artif. Intell., vol. 2009, pp. 4:2–4:2, Jan. 2009.

[16] B. Li, "Cross-Domain Collaborative Filtering: A Brief Survey," in 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, 2011, pp. 1085–1086.

[17] P. Singh and G. J. Gordon, "Relational Learning via Collective Matrix Factorization," in Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 2008, pp. 650–658.

[18] X. Liu, E. Khudkhudia, L. Wen, and V. Sajja, "An Intelligent Computational Argumentation System for Supporting Collaborative Software Development Decision Making," Artificial Intelligence Applications for Improved Software Engineering Development: New Prospects, pp. 167–180, 2010.

[19] S. Sigman and X. F. Liu, "A computational argumentation methodology for capturing and analyzing design rationale arising from multiple perspectives," Information & Software Technology, vol. 45, pp. 113–122, 2003.

[20] X. (Frank) Liu, E. C. Barnes, and J. E. Savolainen, "Conflict Detection and Resolution for Product Line Design in a Collaborative Decision Making Environment," in Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work, New York, NY, USA, 2012, pp. 1327–1336.

[21] M. R. Smith, M. S. Gashler, and T. Martinez, "A hybrid latent variable neural network model for item recommendation," in 2015 International Joint Conference on Neural Networks (IJCNN), 2015, pp. 1–7.

[22] G. Takács, I. Pilászy, B. Németh, and D. Tikk, "Matrix Factorization and Neighbor Based Algorithms for the Netflix Prize Problem," in Proceedings of the 2008 ACM Conference on Recommender Systems, New York, NY, USA, 2008, pp. 267–274.

[23] A. Mnih and R. R. Salakhutdinov, "Probabilistic Matrix Factorization," in Advances in Neural Information Processing Systems 20, J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, Eds. Curran Associates, Inc., 2008, pp. 1257–1264.

[24] S. H. Schwartz, "An Overview of the Schwartz Theory of Basic Values" Online Readings in Psychology and Culture [Online]. Available from: https://scholarworks.gvsu.edu/orpc/vol2/iss1/11