Abstract—This paper presents unsupervised algorithms to uncover polarization in social networks (namely, Twitter) and identify polarized groups. The approach is language-agnostic and thus broadly applicable to global and multilingual media. In cases of conflict, dispute, or situations involving multiple parties with contrasting interests, opinions get divided into different camps. Previous manual inspection of tweets has shown that such situations produce distinguishable signatures on Twitter, as people take sides leading to clusters that preferentially propagate information confirming their individual cluster-specific bias. We propose a model for polarized social networks, and show that approaches based on factorizing the matrix of sources and their claims can automate the discovery of polarized clusters with no need for prior training or natural language processing. In turn, identifying such clusters offers insights into prevalent social conflicts and helps automate the generation of less biased descriptions of ongoing events. We evaluate our factorization algorithms and their results on multiple Twitter datasets involving polarization of opinions, demonstrating the efficacy of our approach. Experiments show that our method is almost always correct in identifying the polarized information from real-world twitter traces, and outperforms the baseline mechanisms by a large margin.

I. INTRODUCTION

This paper presents algorithms to uncover polarization on social media networks, such as Twitter, and identify opposing sets of biased tweets. We define polarization as a condition in which two opposing views enjoy wide support by different groups in a community. For example, a community might become divided over a political or social issue; this is often manifested as opposing views on how the issue should be resolved. Often, the conflict extends to claims about factual observations, such as whether a person had a gun on them or not at a particular time. These widely held and reported conflicting beliefs obfuscate descriptions of the real progression of events as a result of various injected biases. To uncover a less biased (i.e., more neutral) description of events, it is important to identify polarization and distill neutral observations from the reported mix, which motivates the work in this paper.

In this paper, we present a polarization model for information networks, and show that the presence of polarized groups can be detected by considering dependence among posted observations. Using matrix rank as a parameter, we propose a matrix factorization approach to uncover polarization. We explore different degrees of polarization and compare the quality of separation (of tweets of opposing polarity) across different algorithms using real traces collected from Twitter. The work is motivated, in part, by the increased reliance on social networks as news sources. Social network based news dissemination is different from traditional news networks, where raw information goes through curation by expert analysts and journalists before publication. In contrast, in the social media, anybody can post anything. Polarization or bias is inevitable [1]. Hence, tools are needed to clean-up the media before consumption as news.

Our results demonstrate that opposing sets of polarized tweets and sources can be identified automatically (with no content analysis or natural language processing) by the aforementioned matrix factorization approach. Experiments show that the proposed algorithm performs much better than the baseline methods in terms of accuracy in unveiling the polarized sources and groups. The underlying intuition lies in that, in cases of conflict, parties of opposing views tend to disseminate dissimilar sets of claims on social networks. Hence, some of the disseminated tweets can be separated into two subsets propagated by largely non-overlapping sets of sources. We represent the set of tweets as a matrix, where one dimension represents sources and the other represents their tweets (or claims), and where non-zero entries represent who said/forwarded which tweet. Given such a matrix, our algorithm uncovers the underlying latent groups and claims, thereby identifying both the conflicted social groups and their respective views. The language-independent nature of the approach makes it especially advantageous in applications involving multilingual media such as Twitter, since no dependency on a particular lexicon is involved.

It should be noted that the problem addressed in this paper is different from detecting communities in the social network. We observe that in practice, the crawled traces of polarized tweets are often intermixed with a large set of neutral sources and observations. Since neutral observations can also be relayed by polarized sources, and since neutral sources may mix and match view of different polarities, the task of separating the polarized clusters becomes a much harder problem. In this setting, algorithms based on community detection do not correctly identify the polarized clusters. In this paper, solutions are presented that explicitly handle the existence of a neutral population of sources and claims that blurs the boundaries of polarized groups.
The problem addressed in this paper is also different from the commonly addressed problem of veracity analysis on social media. There can be several types of bias present on the social medium. In the extreme case, one or more of the polarized sources are malicious. People post false information to glorify or defame certain acts or causes. This propaganda may result in an ‘online war’ on the social platform, and veracity analysis might be used to detect improbable claims. However, it is often that polarization is more benign. Individuals do not post entirely fabricated observations, but rather color true observations depending on their opinions. A more subtle form of polarization occurs when people selectively propagate or suppress observations based on their bias. For example, a person supporting political party X may only forward (true) positive information about X and the negatives about Y. Another person can forward (true) information about the opposite. In this case, veracity analysis does not help identify polarization.

Also, note that the problem is different from sentiment analysis. A statement that mentions, say, a president and features a negative sentiment might not actually be opposing the president. It might be negative on something else. For example, consider these tweets regarding a former Egyptian president (Morsi): “Saudi Arabia accused of giving Egypt $1B to oust Morsi” or “Egypt clashes after army fire kills #Morsi supporters”. Both tweets mention “Morsi” (the president) and feature negative sentiments (due to use of such keyword as “accuse”, “oust”, “clash” and “kill”). However, reading them carefully, it is easy to see that both sympathize with the president depicting him and his supporters as victims.

The paper is therefore novel in addressing the problem of identifying and separating polarization as opposed to, for example, performing veracity analysis, community detection, or sentiment analysis.

The rest of this paper is organized as follows. Section II illustrates a motivating example from Twitter that leads to our problem, models polarization in social network, and formulates the problem. Section III derives a matrix factorization based gradient descent algorithm to estimate the polarities. In section IV we describe the implementation, evaluate our algorithm, and compare it to other baselines. Section V reviews the literature related to polarization in social networks. The paper concludes with a discussion in section VI.

II. Polarization in Social Networks

The end result (of identifying polarization), presented in this paper, could in principle be accomplished using semantic analysis and natural language processing. The goal of our work, however, is to achieve that end in a language agnostic manner. There are two reasons why this is important. First, on a multi-lingual, multi-national medium, such as Twitter, the number of languages used is large. Developing a model for each language specifically to identify polarization is a rather expensive undertaking. Second, it is not always clear that understanding the language helps understand the polarity of a statement. Consider, for example, the following tweet about Jamala, the winner of the Eurovision competition in 2016: “Jamala performs Bizim Qirim at Kiev concert Hall, 18 May, 2015. The same song wins Eurovision one year later”. Is this tweet advertising Jamala (i.e., is “pro”) or is it against her (i.e., is “anti”)? Someone not familiar with the underlying background might consider it pro. In reality, it is not. Eurovision rules dictate that Eurovision songs have to be original. By claiming that the song was performed a year earlier, the source suggests that the entry should have been disqualified. The need to understand situation-specific context on a case-by-case basis poses significant challenges when it comes to building general-purpose schemes for identifying polarity.

Our approach uses a different intuition. Individuals retweeting statements such as the above, on average, understand their context and polarity. Their behavior reflects their understanding. Hence, by monitoring such collective behavior (namely, the overall propagation patterns of tweets), and clustering it by similarity, it is possible to separate “pro” versus “anti” without having to understand the language. In essence, we harness the collective intelligence of the social medium. In the following sections, we introduce the information model for polarized social networks, and formally define the problem.

A. Information Model for Social Networks

Online social platforms often allow mechanisms to crawl public information. The crawled information at first goes through domain specific cleaning or filtering steps. The content is then clustered using appropriate similarity measurement, which helps to consolidate small variations in the data, and generate a rich information network. A cluster of the very similar observations is considered as a single assertion, and the people or the authors who posted those observations are considered as sources. The bipartite graph from the sources to the assertions is called a source-assertion network. The method of generating this network from the crawled data has been discussed in detail in different works [2], [3]. In this paper, we represent the source-assertion network as a binary source-assertion matrix $A$ of dimensions $s \times c$, where $s$ is the number of sources, and $c$ is the number of assertions. If source $i$ claims assertion $j$, then $a_{ij} = 1$, otherwise $a_{ij} = 0$.

In addition to the source-assertion network, a social influence or dependency network can also be derived (or crawled), where an $(s, t)$ edge denotes that source $s$ has a tendency to forward information if it is received from source $t$. This graph can be weighted when the intensity of influence or dependency is considered into the model, or it can be simplified as an unweighted graph of binary relations. We represent it as a $s \times s$ social dependency matrix $T = [t_{ij}]$ of binary values. It can be derived from an explicit social network such as Twitter follower-followee relations. It can also be estimated from retweet behavior of the sources. Netrapalli and Sanghavi [4] model the propagation of information through the social network as cascades of epidemics. Given the tweets along with sources and timestamp information, they solve the inverse problem of finding the latent propagation structure. In
this paper, we estimate the social dependency network using their maximum likelihood estimation mechanism.

B. Modeling Polarized Information Networks

In this section we augment the source-assertion network with additional states for polarized scenario. Please note that these models are developed according to the real-world observations reported by multiple independent works [5], [6].

We define a polarity group as the set of different senses (polarities) relative to a polarity context (pole). For example, in a US political parties context, the polarity group can be \( K = \{ \text{democrat, republican} \} \). Please note that although two polarities are common, the polarity group can contain more than two members, if the context is not bipartite. For example, the polarity group of the former example could also be \( K' = \{ \text{democrat, libertarian, republican} \} \). Given a set of assertions strictly related to the polarity context, each assertion can be classified as one of the polarities from the polarity group. However, due to the nature of data collection, often there are assertions that do not belong to any of the polarities, which can be termed as neutral, or nonpolarized assertions. For example, every tweet that contains the keyword Morsi is not necessary pro-Morsi or anti-Morsi.

Obtaining the ground truth about the polarity of an assertion requires human effort (that we want to automate). It requires a human grader to understand the content of the assertion and its context. Then the grader assigns a polarity from the polarity group, or classifies it as a neutral or nonpolarized assertion. A source is polarized if its odds of making non-neutral claims of a particular polarity is above a threshold \( \tau \), otherwise a source is neutral.

Suppose the polarity group is \( K = \{ \text{pro, anti} \} \). In that case, the bipartite source-assertion network takes the form shown in Figure 1a. Circles \( S_1 - S_6 \) represent six sources, and squares \( C_1 - C_7 \) represent seven assertions. Empty circles (squares) represent neutral sources (assertions). Filled circles (squares) represent polarized sources (assertions). Arrows represent claims of different polarities. The relationship between the polarized and the neutral components can be represented as Figure 1b. Here sources (assertions) with particular polarities are consolidated together as a single circle (square) representing that polarity. The rest of the vertices are consolidated as neutral sources and neutral assertions. The polarized vertices (pro and anti) are further consolidated as the polarized network.

C. Problem Formulation

Consider a scenario with two polarities, namely pro and anti. Some of the observations posted in the social media favors or averts the pro camp. Some of the observations do the same for the anti camp. In this paper, our goal is to separate the polarities. To develop the formulation, we consider the simplified case with opposing polarities only, without the presence of the neutral network. Later we show how the solution to the simplified formulation is adapted to solve the general case with a huge neutral network obscuring the polarized network.

The observation that there are polarized factions that do not share posts contradicting their polarities, allows us to separate them. Consider \( \Pr(S, C_j) \), probability that source \( i \) shares an observation \( j \). \( \Pr(C_j^q) \) denotes the probability that assertion \( j \) is of polarity \( q \). \( \Pr(S_i^{\text{pro}}) \) denotes the probability that source \( i \) is of polarity pro. We can then write equation 1.

\[
\Pr(S, C_j) = \Pr(S, C_j^{\text{pro}}) \cdot \Pr(C_j^{\text{pro}}) + \Pr(S, C_j^{\text{anti}}) \cdot \Pr(C_j^{\text{anti}})
\]

When the assertion opposes the polarity of a source, the source is not going to share it. Therefore, both \( \Pr(S, C_j^{\text{pro}}) \) and \( \Pr(S, C_j^{\text{anti}}) \) reduces to 0.

\[
\Pr(S, C_j) = \Pr(S, C_j^{\text{pro}}) \cdot \Pr(C_j^{\text{pro}}) + \Pr(S, C_j^{\text{anti}}) \cdot \Pr(C_j^{\text{anti}})
\]

Now we consider the terms \( \Pr(S, C_j^{\text{pro}}) \) and \( \Pr(S, C_j^{\text{anti}}) \) in equation 2. In an ideal situation, these values are 1, making a source share each and every observation whenever the polarity matches. In practice, this does not happen. \( \Pr(S, C_j^{\text{pro}}) \) and \( \Pr(S, C_j^{\text{anti}}) \) depends on various social and human factors, but we can simplify this probability as a combination of two independent components, (i) activity level of the source \( i \) denoted by \( \text{act}(S_i) \), and (ii) circulation level of the assertion \( j \) denoted by \( \text{cir}(C_j) \). Taking \( \delta \) as a scaling constant, we can write \( \Pr(S, C_j^{\text{pro}}) = \delta \cdot \text{act}(S_i) \cdot \text{cir}(C_j) \).

\[
\Pr(S, C_j^{\text{pro}}) = \delta \cdot \text{act}(S_i) \cdot \text{cir}(C_j)
\]

Consider the general case with \( k \) polarities, \( q \in \{1..k\} \). \( U = [u_{iq}] \) is an \( s \times k \) matrix, and \( V = [v_{jq}] \) is a \( c \times k \) matrix. Activity levels of the sources and their probabilities to belong to particular polarized camps are represented in \( U \). Circulation levels of the assertions and their probabilities

\[
\text{Pr}(S, C_j^{\text{pro}}) = \delta \cdot \text{act}(S_i) \cdot \text{cir}(C_j)
\]

Consider the general case with \( k \) polarities, \( q \in \{1..k\} \). \( U = [u_{iq}] \) is an \( s \times k \) matrix, and \( V = [v_{jq}] \) is a \( c \times k \) matrix. Activity levels of the sources and their probabilities to belong to particular polarized camps are represented in \( U \). Circulation levels of the assertions and their probabilities

\[
\text{Pr}(S, C_j^{\text{pro}}) = \delta \cdot \text{act}(S_i) \cdot \text{cir}(C_j)
\]
to favor particular camps are represented in \( V \). Therefore, \( u_{iq} = \delta_{i,act}(S_i) \Pr(S_i^T) \), and \( v_{jq} = \delta_{j,ci}(C_j) \Pr(C_j^T) \). If \( A = [\hat{a}_{ij}] \) represents the probability of a source to share a particular assertion, we can rewrite equation 3 as \( \hat{a}_{ij} = \sum_{q=1}^{k} u_{iq}v_{jq} \), or \( \hat{A} = UV^T \).

Given \( A = [\hat{a}_{ij}] \) as the actual observations on whether source \( i \) shared assertion \( j \) in the social network, \( T = [t_{ij}] \) as the social dependency matrix on whether source \( i \) is likely to forward information received from source \( j \), and a polarity group \( K \), our goal is to estimate \( U \) and \( V \) component matrices that allow us to separate the polarized components.

D. Solution Approach

Given \( k \) as the rank, we can factorize \( A \) to estimate \( U \) and \( V \) components. Please note that, \( A = UV^T = URR^{-1}V^T = (UR)(VR^{-T})^T \), where \( R \) is a \( k \times k \) multiplier matrix. Therefore, factorizing \( A \) without any constraint will result in \( UR \) and \( VR^{-T} \) as component matrices. In the following section, we add appropriate constraints to limit the arbitrariness of \( R \). In the simplified case, when there is only the polarized network, rank of \( A \) is exact. Sources and assertions of different polarities can be uniquely separated using the estimated factor matrices \( U \) and \( V \). However, in the presence of a large neutral network, the number of polarized camps \( k \) does not correctly represent the rank of \( A \). In this case, different separations are possible that can approximate the observation matrix \( A \). We estimate multiple instances of \((U, V)\) using different initializations. For each instance, observations are partitioned into different polarities. Instances are generally related to each other in terms of similarity between corresponding partitions. Anomalous instances that are highly different than the rest are discarded. Rest of the instances are aggregated to estimate the final partitions.

III. A Matrix Factorization Approach to Uncover Polarization

In this section, we derive a gradient-descent algorithm to jointly estimate the polarization of the sources and assertions. Suppose \( A \) is the \( s \times c \) source-assertion matrix. Polarization of the sources and the assertions can be estimated from \( A \) by factorizing it in the form of matrices \( \hat{U} \) and \( \hat{V} \), defined earlier.

If \( k = \text{rank}(A) \), \( A \) can be factorized exactly in the form \( A = UV^T \), where \( U = [u_{ij}] \) is an \( s \times k \) matrix that represent the polarization of the sources, and \( V = [v_{ij}] \) is a \( c \times k \) matrix representing the polarization of the assertions. Please note that \( A \) is an incomplete matrix because when source \( i \) does not claim assertion \( j \), it can be that source \( i \) did not have opportunity to observe \( j \), or \( i \) ignored assertion \( j \) after observing it. Therefore a sample of the missing edges in the source-assertion network are represented as \( a_{ij} = 0 \), and the rest are considered as missing. Because \( A \) is incomplete, we do not know the exact rank of \( A \). However, as visible from Figure 1a, the sources and assertions of different polarized camps are independent when they are sharing information related to the particular polarized scenario. Hence, we take the number of polarized groups \( |K| \) as \( \text{rank}(A) \), and approximately factorize \( A \) as \( A \approx UV^T \).

Note that this condition is defined only for the entries of \( A \) that are observed. Therefore, let us define the set \( O = \{ (i, j) : a_{ij} \text{ is observed} \} \) to be all the indices in matrix \( A \) that are observed. Given a particular \( U \) and \( V \), the estimate of an entry of \( A \) is given by \( \hat{a}_{ij} = \sum_{q=1}^{k} u_{iq}v_{jq} \). Therefore, the estimation error \( e_{ij} \) is: \( e_{ij} = a_{ij} - \hat{a}_{ij} \). In order to approximately factorize the matrix \( A \), we would like to minimize the objective function, which is equal to the sum-of-squared errors \( J = \sum_{(i,j) \in O} e_{ij}^2 = \sum_{(i,j) \in O}(a_{ij} - \sum_{q=1}^{k} u_{iq}v_{jq})^2 \). This form of the objective function, however, can result in infinitely many solutions, each of which minimizes \( J \). We impose the following constraints. These constraints correspond to overfitting of the objective function, and impact of the social dependency matrix on polarization consistency.

1) Regularization: If \( U \) and \( V \) is a particular solution, then multiplying \( U \) by an arbitrary \( k \times k \) real matrix \( R \), and multiplying \( V \) by \( R^{-T} \) would also minimize \( J \), provided \( R \) is invertible. This is because \( UV^T = UV^TRR^{-1}V^T = U(RR^{-1})V^T = (U R)(VR^{-T})^T \), where \( I \) is a \( k \times k \) identity matrix. Depending on the chosen initial values or the missing entries, the objective function can overfit the model, or oscillate between multiple solutions. Therefore, we impose a regularization constraint on \( J \). We choose to use \( L_2 \)-regularization \( \lambda(||U||_F^2 + ||V||_F^2) \) so that arbitrarily large values in \( R \) would be prevented. The value of \( \lambda > 0 \) represents the value of the regularization parameter.

2) Social dependency-based polarization consistency: We observed that polarization in the crawled data is obscured by a large nonpolarized or neutral network. Presence of such sources and assertions result in multiple separations between the different polarity groups likely. The objective function would result in multiple candidate solutions. Therefore, we add an additional constraint. Users that depend on one another according to the social dependency matrix \( T \) are more likely to exhibit polarization consistency. So the columns in \( \hat{U} \) corresponding to sources who depend on each other contains similar entries. If \( \pi_s \) is the row in \( \hat{U} \) corresponding to source \( x \), the additive component \( \gamma_i \pi_s - \pi_{\text{act}} \) would add a penalty whenever source \( i \) depends on source \( j \), but their corresponding columns vary. Here \( \gamma > 0 \) is a parameter that regulates the importance of the social consistency component. This parameter can be chosen later in the tuning phase. Please note that adding this constraint will increase the error in the factorization but it will favor solutions that have higher consistency with the social dependency network.

Therefore, by adding these terms, our objective function becomes \( J = \sum_{(i,j) \in O}(a_{ij} - \sum_{q=1}^{k} u_{iq}v_{jq})^2 + \sum_{i,j} \gamma_i \pi_{s} - \pi_{\text{act}} ||^2 + \lambda(||U||_F^2 + ||V||_F^2) \), which needs to be minimized.

A. Solving the Optimization Problem

We minimize \( J \) with respect to the parameters in \( U \) and \( V \) using gradient descent method. We rewrite the objective
1: procedure FACTORIZE(A, T, k)  
2: Randomly initialize \( U, V \)  
3: repeat  
4:     for each \((i, q)\) do  
5:         \( u_{iq} \leftarrow u_{iq} - \alpha \frac{\partial J}{\partial u_{iq}} \)  
6:     end for  
7:     for each \((j, q)\) do  
8:         \( v_{jq} \leftarrow v_{jq} - \alpha \frac{\partial J}{\partial v_{jq}} \)  
9:     end for  
10:     for each \((i, q)\) do  
11:         \( u_{iq} \leftarrow u_{iq} \)  
12:     end for  
13:     for each \((j, q)\) do  
14:         \( v_{jq} \leftarrow v_{jq} \)  
15:     end for  
16: until convergence reached on \( U, V \)  
17: return \((U, V)\)  
18: end procedure

Fig. 2. Gradient descent algorithm for factorization function, and compute the partial derivative of \( J \) with respect to each parameter in \( U \) and \( V \):

\[
J = \sum_{(i,j) \in O} e_{ij}^2 + \sum_{i,m} \gamma t_{im} \sum_{q=1}^k (u_{iq} - u_{mq})^2 \\
+ \lambda (||U||_F^2 + ||V||_F^2)
\]

\[
\frac{\partial J}{\partial u_{iq}} = 2 \sum_{j:(i,j) \in O} e_{ij} (-v_{jq}) \\
+ 2 \sum_{m} (\gamma t_{im} + \gamma t_{mi}) (u_{iq} - u_{mq}) + 2\lambda u_{iq}
\]  

\[
\frac{\partial J}{\partial v_{jq}} = 2 \sum_{i:(i,j) \in O} e_{ij} (-u_{iq}) + 2\lambda v_{jq}
\]

Note that we can ignore the constant factor of 2 throughout the RHS of the aforementioned equation for the purposes of gradient descent. We compute all partial derivatives with respect to the different parameters in \( u_{iq} \) and \( v_{jq} \) to create gradient matrix \( \nabla U \) of dimensions \( s \times k \), and \( \nabla V \) of dimensions \( c \times k \). The gradient-descent method updates \( U \leftarrow U - \alpha \nabla U \), and \( V \leftarrow V - \alpha \nabla V \), where \( \alpha \) is the step-size. The parameter \( \gamma \), \( \lambda \) can be selected using cross-validation. Figure 2 enumerates this mechanism.

We impose the additional constraint that the entries of the matrix \( U \) and \( V \) are non-negative, although the optimization objective function remains the same. It provides a sum-of-parts decomposition to the source-assertion matrix as dictated by the problem formulation. To achieve this, during initialization, the entries of matrices \( U \) and \( V \) are set to non-negative values in \((0, 1)\). During an update, if any entry in \( U \) or \( V \) becomes negative, then it is set to 0.

**B. Separating Polariities using \( \hat{U} \) and \( \hat{V} \)**

Activity levels of the sources and their probabilities to belong to particular polarized camps are represented in \( U \). Circulation levels of the assertions and their probabilities to favor particular camps are represented in \( V \). Rows of \( U \) and \( V \) can be considered as points in a \( k \)-dimensional euclidean space. In the simplified case, where the source-assertion matrix consists of only the polarized network with \( K = \{\text{pro, anti}\} \), the extreme points of \( U \) or \( V \) are \((1, 0)\) or \((0, 1)\). These points represent the sources making all the \text{pro} assertions, or making all the \text{anti} assertions, respectively. All the other points would fall on either \( x \)-axis or \( y \)-axis. However, in the general case, the neutral network is present, hence it is possible to have points that fall within the right triangle defined by vertices at \((0, 0), (1, 0), \) and \((0, 1)\).

Through factorization we have estimated \( \hat{U} = UR \), and \( \hat{V} = VR^{-T} \). This multiplier \( R \) causes the estimated values in \( \hat{U} \) or \( \hat{V} \) to have been applied a linear transformation. A linear transformation in general can be decomposed to several rotations and scales. Due to the constraints we have added to \( J \), effect of \( R \) is small. Figure 3a shows an output where the rows of \( \hat{V} \) are plotted on the 2D plane for a particular experiment. We observe that the multiplier \( R \) has been mostly restricted to a diagonal matrix corresponding to scale transformation.

Figure 3a also plots the ground truth of the assertions as obtained via manual annotation. To separate the different polarity groups, we note that linear transformations preserve parallel lines. Therefore, the midpoint of a transformed line corresponds to the transformation of the midpoint of the original line. We can separate the polarities by finding the pair of assertions \((a, b)\) from \( \hat{V} \) with maximum euclidean distance, i.e. \( \arg \max_{a,b} ||v_a - v_b||^2 \), and assigning the other assertions to either the polarity of \( a \) or \( b \), using a nearest neighbor rule. However, we observe that \( R \) has been mostly restricted to scaling. Therefore, to obtain a separation of the polarities, assertion \( j \) can be assigned to the group corresponding to \( \arg \max_q \{\hat{v}_{jq}\} \). For Figure 3a, this corresponds to using sign of \( \hat{v}_{j,1} - \hat{v}_{j,2} \) as the separator. The sources can also be separated in a similar manner using \( \hat{U} \).

**C. Ensemble of Factorization Experiments**

We note that in the presence of a large neutral network, different runs of factorization results in different separations. Figure 3b illustrates the receiver operating characteristics (ROC) for the egypt scenario. ROC curve plots true positive rate vs. false positive rate, and is used to assess the quality of classification. The optimal algorithm has an area of 1 under
the ROC, which happens when the output includes all the true positives before any of the false positives.

Figure 3b plots the distribution of true positive rate for different false positive rates, and shows that although the factorization algorithm is able to achieve good performance, there is significant variance in the separation obtained from the results. We also compare the result of when the social dependency network is used as a constraint vs. when it is not. We observe that although use of social dependency network improves the quality of the results, there is still variance in the separation. We, therefore, use an ensemble of factorization experiments to estimate the most likely assignments of the assertions to the respective polarities.

It is not possible to directly compare \( \hat{V}_m \) with \( \hat{V}_n \), when \( m \) and \( n \) different experiments, because of the transformation difference caused by \( R_m \) and \( R_n \). We, therefore, separate the assertions to different polarity groups for each experiment. Experiments are aligned to each other using a mechanism based on Jaccard distance [7]. We explain it for two polarities, i.e. \( k = 2 \). Figure 4 shows the algorithm, with the procedure \textsc{EstimatePolarities} at line 38 being the starting point. Suppose the separation generated by factorization experiment \( m \) is \( B^1_m \) and \( B^2_m \), and the separation generated by factorization experiment \( n \) is \( B^1_n \) and \( B^2_n \). It is possible that \((B^1_m, B^2_m)\) aligns with \((B^1_n, B^2_n)\), or with \((B^2_n, B^1_n)\). Figure 5a illustrates the two cases considering the polarities as \texttt{pro} and \texttt{anti}. We compute a 2 \( \times \) 2 matrix of the Jaccard distances between the separations created by the experiments. Jaccard distance between two sets \( X, Y \) is defined as \( 1 - \frac{|X \cap Y|}{|X \cup Y|} \). It is used to assess how similar or dissimilar they are. In order for the two experiments to match, either the main diagonal will exhibit more similarity than the anti-diagonal (or vice versa). If the maximum in the matching diagonal is below a threshold \( \tau_{\text{edge}} \), given their difference is within \( \tau_{\text{diag}} \), the experiments are considered to match and a weighted edge is added to \( G \), the graph of experiments. The weight is considered positive if the experiments matched along the main diagonal, and negative if the experiments matched along the anti-diagonal. Figure 5b shows an experiment graph (without the weights) obtained from 20 experiments on the egypt polarized scenario. Experiments that highly differ from the others remain isolated in the experiment graph, or form small islands. We find the experiment with the largest degree in \( G \), and aggregate all the adjacent experiments.

There can be several procedures to aggregate the experiments. We keep a vector of frequencies \((x_j, y_j)\) for each assertion. \( x_j \) and \( y_j \) counts the number of times assertion \( j \) has been assigned to polarity \( x \) or \( y \). Normalizing these frequencies and sorting them by the difference of the vector components \((x_j - y_j)\) gives us a spectrum of assertions, from the most likely to belong to one polarized camp to the most likely to belong to the other camp.

IV. EVALUATION

We evaluate our algorithm in the context of polarized scenario in Twitter. Tweets were crawled in real time with tools using Twitter search API. Three sets of traces were collected that contains polarization around (i) former Egyptian president Morsi, (ii) Eurovision song contest 2016 winner Jamala, (iii) US Presidential election candidate Donald Trump.
The entire collection of recorded traces was clustered based on text similarity to generate a representative summary [3], [7], [8]. We implemented the factorization program using Java. Sparse matrix data structures were used to efficiently store large matrices. Different components of the pipeline were interfaced using Python. Factorization was performed followed by the ensemble of multiple experiments to separate the tweets between two polarities. We used \( k = 2 \), \( \alpha = 0.001 \), \( \gamma = 0.1 \), \( \lambda = 0.5 \), ensemble size \( = 20 \), \( \tau_{\text{diag}} = 0.15 \), \( \tau_{\text{edge}} = 0.7 \). We compare the quality of separation obtained by our algorithm with the following related techniques:

1) **Sentiment Analysis:** Sentiment analysis [9], [10] uses language models to understand the sense of content written in natural language, and classifies them as having positive, negative, or neutral sentiment. To annotate the assertions we used Umigon [11] and Sentiment140 [12]. These are freely available specialized tools for performing sentiment analysis on tweets.

2) **Community Detection:** Polarized sources are unlikely to share tweets contradicting their own polarity. Therefore detecting communities in the social network is a candidate mechanism for separating the polarities. We partition the graph of sources and assertions into \( k = 2 \) communities with the objective of minimizing the edge-cut (number of edges that cross partitions). We used Metis [13] to obtain that. In addition to detecting communities, we have added another baseline where the assertions in each community are ranked by their degrees. We refer this mechanism by MetisVoting.

3) **Veracity Analysis:** Algorithms to perform veracity analysis [2], [14]–[18] utilize the source-assertion network to uncover likely facts from the set of tweets. They can be considered related techniques if one of the polarities have more affinity towards factual information. We used the EM-Social [2] algorithm to jointly assess the credibility of the sources and the assertions.

### A. Egypt

Mass street protests against the then president Mohamed Morsi was followed by a coalition led by the army chief [19] on July 3, 2013. The president was deposed and arrested by the army along with other leaders of his political party. This incident resulted in protests and clashes between the supporters and the opponents of the removed president. Tweets related to the deposed president were collected. For the purpose of evaluation, the largest 1000 clusters containing English tweets were read and manually annotated on whether they were pro-Morsi, anti-Morsi, or neutral in sense. There were 199 pro-Morsi, 109 anti-Morsi, and 692 nonpolarized assertions.

Figure 6a compares the receiver operating characteristics (ROC) achieved from our algorithm to other baselines. To obtain the ROC, the set of assertions were sorted in the order of highest polarity in one class to the highest polarity in the other class. When consuming the assertions in that sequence, finding a pro-Morsi assertion was considered as an occurrence of true positive, and finding an anti-Morsi assertion was considered as an occurrence of false positive. The area under ROC curve measures how well an algorithm performs both in terms of finding the correct answers, and omitting the wrong answers.

Factorization algorithm performs really well. Area under the ROC curve is approximately 0.93. Both Umigon and Sentiment140 performed just as good as a random technique, because (i) a large number of assertions were classified as neutral, and (ii) as described earlier in the paper, sentiment analysis is not the correct technique to uncover polarization. An assertion having positive sentiment can be a positive statement favoring either camp. Hence, sentiment is orthogonal to polarity. EM-Social is also unable to differentiate between the polarities. It illustrates that there was almost no correlation between the veracity of a tweet and any particular polarity. Metis and MetisVoting techniques performed better than the other baselines because of their graph partitioning nature. However, the source-assertion network had around 80% nonpolarized sources and 70% nonpolarized assertions. Therefore a community detection analysis was unable to perform well.

Table 1 shows the top 5 tweets from each polarity from the separation achieved using our algorithm. Note that the tweets on the left column sympathize with the deposed president or his supporters. On the other hand, the tweets on the right column is vocal against the deposed president and his political party, and reporting negative news about them.

### B. Eurovision

Susana Jamaladinoa (Jamala) from Ukraine was the winner of Eurovision 2016, an annual European song competition. It was unexpected to many as the expected winner was Russia or Australia according to pre-competition polls. The winning song, 1944, according to the artist, was telling a personal story
related to her family in the aftermath of the deportation of the Crimean Tatars by the Soviet Union. However, it was also alleged to have political connotations against Russian interference with Crimea in 2014. Tweets related to Jamala were collected for five days after her win. The largest 1000 assertions were manually annotated. There were 600 pro-Jamala, 239 anti-Jamala, and 161 neutral assertions.

Figure 6b compares quality of factorization with other baselines. Our algorithm performs best in this scenario. Metis performs reasonably better than the earlier case because of relatively better community separation. Because there were many tweets with positive sentiment that were correlated to pro-Jamala, Umigon also performed better than it did in the other cases. Table II shows the top 5 tweets from each polarity from the Facebook dataset. The table shows the top 5 tweets from each polarity from the Facebook dataset.

C. Trump

Donald Trump is the Republican Party nominee for President of the United States in the 2016 election. There have been much debate and controversies around the candidate and his speeches. Tweets were collected using a single keyword Donald Trump, during April 2016. Collected tweets show the radical support by the pro-Trump polarity and the negative opinions or mockery posted by the anti-Trump polarity. For the purpose of generating the ROC curves, the largest 1000 assertions were manually annotated. There were 372 pro-Trump, 522 anti-Trump, and 106 neutral assertions.

Figure 6c compares quality of factorization with other baselines. In this particular scenario, performance of our algorithm is around 2% better than community detection. This is because the corresponding source-assertion network had strong community separation, with only 10% nonpolarized assertions being lightly connected. EM-Social also performs reasonably to find the separations because of the same reason. Table III shows the top 5 tweets from each polarity from the separation achieved using our algorithm. Note that the tweets on the left column are strongly pro-Trump in nature and describing support for him or praising him. On the other hand, tweets on the right column are sharing the negative information about the candidate, and pointing out the controversies.

V. RELATED WORK

Presence of polarization in social networks has been studied in various contexts. Conover et al. [1] study retweet-based social networks and mention-based social networks in political contexts related to U.S. congressional elections. Guerra et al. [20] study polarization metrics for social networks. They
argue that modularity is not directly applicable as a measure of polarity because even without polarization modular communities are present. Uncovering polarization in social networks is important in various contexts. Bakshy et al. [5] study polarization in the context of Facebook. Amin et al. [6], Kase et al. [21] study crowd-sensing and fact-finders in the context of war and conflict situations. In this paper, we solve the orthogonal problem of separating the polarity classes.

Polarization in social network can be viewed as a community detection or graph partitioning problem [22], [23]. We do not directly apply such techniques because of the presence of neutral sources and assertions. Moreover, the requirement of fusing multiple signals to converge to an expected solution required an optimization framework. Sentiment analysis [9], [11] can also be viewed as a related technique to uncover polarization. However, in our case sentiment analysis is not directly applicable because the positive and negative classes in sentiment analysis can be orthogonal to the polarization group in question. Moreover, sentiment analysis is a supervised technique, while our technique is unsupervised. Sentiment analysis can require training and language model to map the sense of the text to sentiments. Even after training, such techniques can miss the assertions that are effectively neutral in sense, but expresses a polarity towards certain object. On the other hand, our method looks at the source information and exploits the network structure to uncover polarity. As it does not consider text information, assertions that are not well connected in the network can be misclassified.

Finding a social-influence network, or source-dependency network has been studied in prior literature [24], [25]. Netrapalli and Sanghavi [4], Myers and Leskovec [26], and Rodriguez et al. [27] use the concept of epidemic cascades to estimate a social network. In this paper, we use the maximum likelihood approach proposed by Netrapalli and Sanghavi [4] to generate the social dependency matrix used as an input to our algorithm.

In addition, ensemble learning [28] is an important research area in machine learning community. Bagging [29] and boosting [30] are two main solutions to ensemble multiple learners. In this paper, we follow this idea by filtering out bad separations by identifying Jaccard distance among the candidates and bagging filtered candidates to generate final polarization result.

VI. CONCLUSIONS

In this paper, we have presented a matrix factorization and ensemble based gradient descent algorithm to uncover polarization in social networks. We have evaluated our algorithm in the context of ongoing disputes, conflicts, or controversies as polarized situations. Experiments show that it can separate the tweets of different polarities by looking just at the source-assertion network and the social dependency network, and can be more than 90% accurate. Our algorithm performs much better than supervised techniques like sentiment analysis. Moreover, it also performs around 20% – 30% better than the community detection approaches, when the separation between

the sources or the assertions of different polarities is obscured because of the presence of a large neutral network. If a particular source or assertion is not well connected to the network, the method can misclassify. Correctly estimating such cases with the help of additional information, deriving confidence bounds for the detected polarity, and jointly estimating polarity of the tweet with its veracity will be addressed in future works.

REFERENCES


