International Journal of Electrical and Computer Engineering (IJECE)
Vol. 7, No. 1, February 2017, pp. 330~336
ISSN: 2088-8708, DOI: 10.11591/ijece.v7i1.pp330-336

Sensing Trending Topics in Twitter for Greater Jakarta Area

Angga Pratama Sitorus, Hendri Murfi, Siti Nurrohmah, Afif Akbar
Departement of Mathematics, Universitas Indonesia, Depok 16424 - Indonesia

Article Info

Article history:

Received Aug 1, 2016
Revised Oct 21, 2016
Accepted Dec 5, 2016

Keyword:

Coherence
NMF
Topic detection
Topic sensing
Twitter

ABSTRACT

Information and communication technology grows so fast nowadays, especially related to the internet. Twitter is one of internet applications that produce a large amount of textual data called tweets. The tweets may represent real-world situation discussed in a community. Therefore, Twitter can be an important media for urban monitoring. The ability to monitor the situations may guide local government to respond quickly or make public policy. Topic detection is an important automatic tool to understand the tweets, for example, using non-negative matrix factorization. In this paper, we conducted a study to implement Twitter as a media for the urban monitoring in Jakarta and its surrounding areas called *Greater Jakarta*. Firstly, we analyze the accuracy of the detected topics in term of their interpretability level. Next, we visualize the trend of the topics to identify popular topics easily. Our simulations show that the topic detection methods can extract topics in a certain level of accuracy and draw the trends such that the topic monitoring can be conducted easily.

Corresponding Author:

Angga Pratama Sitorus,
Departement of Mathematics,
Universitas Indonesia,
Depok 16424 - Indonesia,
Email: apsitorus@sci.ui.ac.id

1. INTRODUCTION

Development of Internet technology has made the spread of information has increased significantly. One of the Internet applications that support the spread of such information is social media [1], [2]. In social media, users not only as consumers of information, but they can act as producer of information also. One of the popular social media for the spread of information is Twitter. Twitter facilitates users to send and read text-based information known as *tweets*. Along with active users producing tweets, then Twitter can be a sensor of real-world events [3]. Another similar use of Twitter is as a sensor of situations in an urban area. The ability to monitor the situations may direct the local government to respond quickly or make public policy. Therefore, Twitter is a potential media for an urban monitoring to support a smart city [4].

Topic detection is the process of determining the topics of a set of textual data [5]. This process is one important step to understanding the tweets which are textual data. Topic detection can be done manually by reading the contents of tweets. However, the manual way is not feasible for large data or a fast response time. Therefore, we need automatic topic detection methods to meet those requirements. One of automatic topic detection methods widely used is *non-negative matrix factorization* (NMF) [6].

Based on Semiocast's report, Jakarta is the most active city in producing tweets[1]. Hence, Twitter may be an important media for urban monitoring in Jakarta area. In this paper, we conducted a study of the application of Twitter as a media for the urban monitoring in Jakarta and its surrounding regions called *Greater Jakarta*. Firstly, we analyze the accuracy of the generated topics in term of their interpretability level. Pointwise mutual information (PMI) is a quantitative measure to calculate the interpretability level [7].

[1] http://semiocast.com/en/publications/2012_07_30_Twitter_reaches_half_a_billion_accounts_140m_in_the_US

By using this measure, we select the optimal number of topics generated for a period. Next, we visualize the frequency of generated topics in time series to show the trend of the topics. This visualization helps us to identify popular topics or extraordinary topics if they exist.

The outline of this paper is as follows: In Section 2, we discuss the research method used in this study. Section 3 describes some results from our simulations. Finally, a general conclusion about the simulations is presented in Section 4.

2. RESEARCH METHOD

The first step in sensing trending topics in Twitter is collecting data by acquisition tweets from Twitter. The acquisition process uses *Twitter API* by streaming all tweets posted in a period. In this paper, we focused on tweets from Jakarta, Bogor, Depok, Tangerang, Bekasi that commonly known as Greater Jakarta. Twitter facilitates the tweet acquisition to a specific location based on geographic coordinates of the location. These coordinates consisted of two points that represent the rectangular shape of the coverage area. Each point represents the two components, namely *longitude* and *latitude*. For the Greater Jakarta area, the first point has latitude 6°1'41.92"S and longitude 106°20'19.34"E, while the second point has latitude 6°41'42.50"S and Longitude 07°19'4.93"E. The two points are shown as two yellow marks in Figure 1.

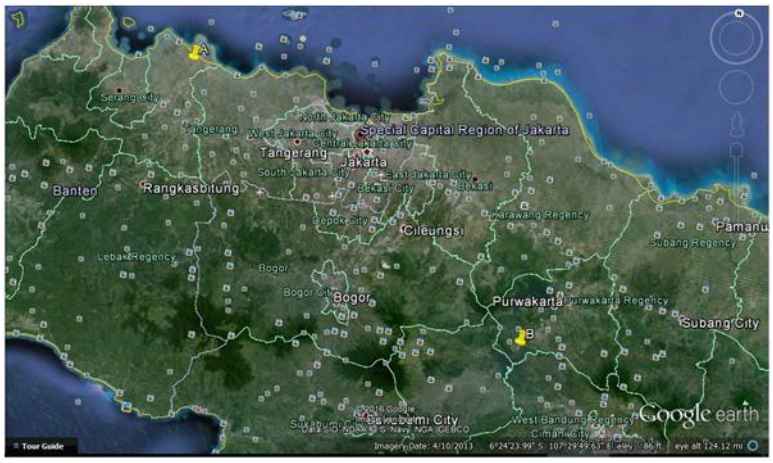


Figure 1. Latitude and Longitude of the Greater Jakarta area

After collecting the tweets, the next process is preparing them by doing feature extraction. The feature extraction process consists of preprocessing, tokenizing, word filtering, and weighting. The preprocessing is changing the text on every tweet into lowercase and deleting every link username. The tokenizing process is separating words in tweets and save them into a word dictionary. The word dictionary that obtained from the previous process will be filtered by *stopwords* to get the most potential words that contribute to the meaning of topics. Each tweet is represented as a vector whose features are words in the dictionary. The weights of the vector are *term frequency inverse document frequency* (*tf-idf*) [6]. *Tf-idf* is defined by

$$f_i(\mathbf{d}) = \frac{tf_i}{df_i}$$

where $f_i(\mathbf{d})$ represents weight of word i on document $\mathbf{d}$, tf_i represent frequency of word i on document $\mathbf{d}$, and df_i represent the number of documents that contain word i. The output of the feature extraction process is a word by tweet matrix A whose columns are tweets.

Non-negative matrix factorization (NMF) is an automatic topic detection method to extracts topics from the word by tweet matrix A. Given the matrix A, NMF method is a matrix factorization method that factorizes the matrix A into two non-negative matrices, i.e. a $m \times k$ matrix W and a $k \times n$ matrix H, such that,

$$A_{mxn} \approx W_{mxk}\, H_{kxn}$$

where m represents the number of words or the dimension of the tweets, and n represents the number of the tweets, and k is the number of topics. According to this formulation, the columns of W represent topics existing in tweets. In other words, a topic is a vector of words. However, it usually uses only top frequent words to describe a topic, i.e. top ten frequent words. While the rows of H represent the association between topics and tweets. If we order the tweets by time then we can describe the trend of each topic using each row of H.

The common method to solve NMF problem is iterative methods by minimizing the differences between A and WH. The NMF problem is formulized in form of *bound-constrained optimization problem* as follow:

$$\min_{W,H} f(W,H) = \frac{1}{2}\|A - WH\|_F{}^2$$
$$\text{s.t}$$
$$w_{ij} \geq 0, h_{kl} \geq 0, \forall i,j,k,l$$

where $\|.\|_F$ is Frobenius Norm. Although the function $f(W,H)$ is convex on the W or H, however, it is not convex on the W and H. Thus, the realistic solution of NMF problem is a local minimum [9]. There are some proposed methods to solve the NMF problem, that is, *projected gradient descent* [10], *multiplicative update rule* [11], and *alternating non-negative least square* (ANLS). From convergence point of view, ANLS gives better convergence condition. The method guarantees that the solutions always converge to the stationary points [9]. In this method, two non-negative least square (NLS) problems are solved in alternating process as shown in Algorithm 1. Therefore, the main problem of the ANLS method is how to solve the NLS problem. There are some methods to solve the NLS problem, i.e. *projected gradient descent* [12], *active-set* [13], and *block principal pivoting* [14]. In our simulation, we solve the NLS problem using the projected gradient descent method. Projected gradient descent algorithm projects a negative value into its nearest non-negative value, that is, 0. Moreover, we initialize W using singular value decomposition based initialization [15]. We use *scikit-learn*, a Python-based library, to implement the methods [16].

Algorithm 1. Alternating non-negative least square algorithm
1. input: A; Output: W, H
2. i = 0
3. inisialisasi $W^i \geq 0$
4. while stopping_criteria = false
5. i = i + 1
6. $H^{(i)} = NLS(A, W^{(i-1)})$
7. $W^{(i)} = NLS(A, H^{(i)})$
8. end while

Before executing an NMF algorithm, we must set the value of the parameter k. The parameter k represents the number of topics to be extracted. How to find the optimal number of topics is the problem in *model selection* of topic detection using NMF. For this task, we calculate the accuracies of topics for some numbers, i.e. 10, 20, …, 100. The number that gives the best accuracy is chosen as the optimal number.

After extracting topics, we calculate the average accuracy of the topics using *pointwise mutual information* (PMI) [7]. PMI shows the coherence values between words in a topic to show the topic's level of interpretability. Let t-th topic of the matrix W consists of n words, that is $(w_{1t}, w_{2t}, \dots, w_{nt})^T$, PMI score of the t-th topic is

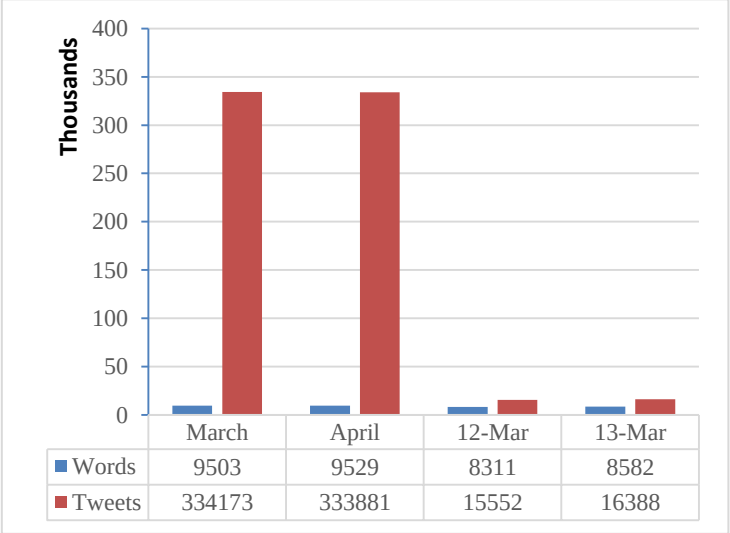
$$PMI(t) = \sum_{j=2}^{n}\sum_{i=1}^{j-1} log\frac{p(w_{it}, w_{jt})}{p(w_{it})p(w_{jt})}$$

where $p(w_{it}, w_{jt})$ is the joint probability of words w_{it} and w_{it} or w_{it} comes up together with word w_j, $p(w_{it})$ and $p(w_{it})$ are the individual probability of word w_{it} and w_{jt}. The higher the PMI score on a topic means more coherent words on the topic. In other words, the interpretation of the topic is easier. We need *a reference corpus* to calculate those probabilities. In our simulation, we use two reference corpuses, that is, *WikiId* which is Indonesian Wikipedia dump from January 2016 and *Berita* which is a collection of Indonesian news. We use an open-source toolkit to calculate the PMI scores[2].

[2] https://github.com/jhlau/topic_interpretability

3. RESULTS AND ANALYSIS

For the simulations, we use two types of data based on a period. The first data is daily data collected on March 12, 2016 and March 13, 2016. The second one is monthly data collected in March 2016 and April 2016. The number of tweets is approximately 334000 tweets per month and 16000 tweets per day. After preprocessing step, the number of words is approximately 9500 words per month and 8500 words per day. The detail statistics of the data is given in Figure 2.



	March	April	12-Mar	13-Mar
Words	9503	9529	8311	8582
Tweets	334173	333881	15552	16388

Figure 2. The Statistics of Data

To find the accuracy of extracted topics for a period, we simulate the accuracies for some numbers of topics, i.e. 10, 20, …, 100. The best accuracy of these numbers of the topic represents the accuracy of the period. Figure 3 and Figure 4 show the accuracy of extracted topics in March 2016 and April 2016, respectively. From both figures, we see that the reference corpus WikiId always give higher PMI scores than the reference corpus Berita for all numbers of the topic. One reason for these results is that the documents of WikiId are longer than ones of Berita. Therefore, the probability that two words exist in a similar document of WikiId is higher than the probability of the words in a similar document of Berita. From Figure 3, we can see that the higher accuracy is achieved when we extract ten topics from tweets in March 2016. It means that the optimal number of topics for that period is ten with a PMI score of 0.43. We can also see from Figure 4 that the higher PMI score is 0.51 in April 2016. This higher score is achieved when we extract ten topics from tweets in that period. This optimal number of topics is similar to the one extracted from the previous month. Hence, ten is the first candidate of the number of topics when we extract topics for next months, especially, when we need a fast respond time in topic detection for the monthly tweets.

Next, we conduct similar simulations for daily data, i.e. March 12, 2016 and March 13, 2016. Figure 5 and Figure 6 visualize the accuracy of extracted topics on March 12, 2016 and March 13, 2016, respectively. From both figures, we see that the reference corpus WikiId still give higher PMI scores than the reference corpus Berita for all numbers of the topic. From Figure 5, we can see that the higher accuracy is achieved when we extract forty topics from tweets on March 12, 2016. These extracted topics have a PMI score of 0.30. We can also see from Figure 5 that the higher PMI score is 0.30 on March 13, 2016. This higher PMI score is achieved when we extract forty topics from tweets in that period. This optimal number of topics is similar to the one extracted from the previous day. Therefore, forty can be the first candidate of the optimal number of daily topics if we do not want to perform model selection in topic detection for the daily tweets.

After estimating the accuracy of the generated topics, the next step is to visualize the trend of the topics. This visualization can use the topic by tweet matrix H which represents the degree of association between topics and tweets. If we assume that tweets have been sorted by time, then we can draw the trend of a topic based on the association values between the topic and each tweet. The higher the association values, then the higher the popularity of the topic. Visualization of those trends allows us to monitor topics for a certain period.

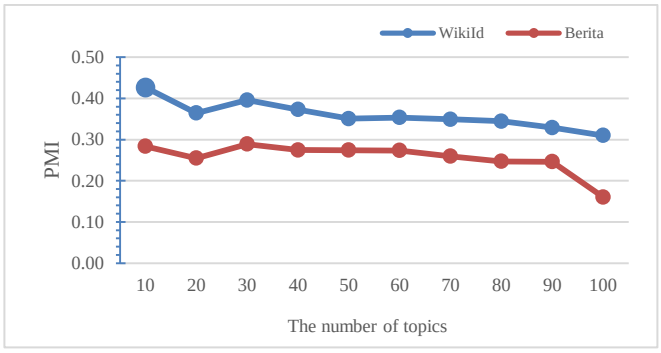


Figure 3. The Accuracy of Topics Extracted in March 2016

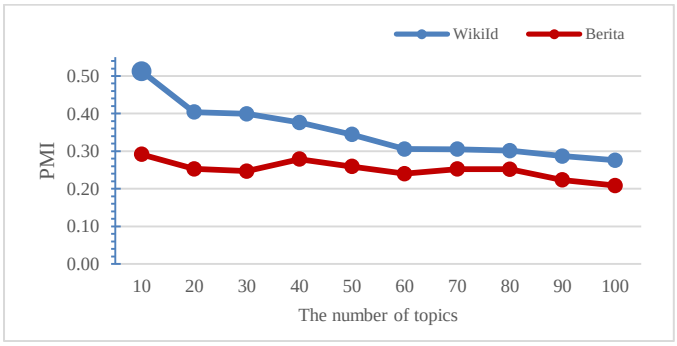


Figure 4. The Accuracy of Topics Extracted in April 2016

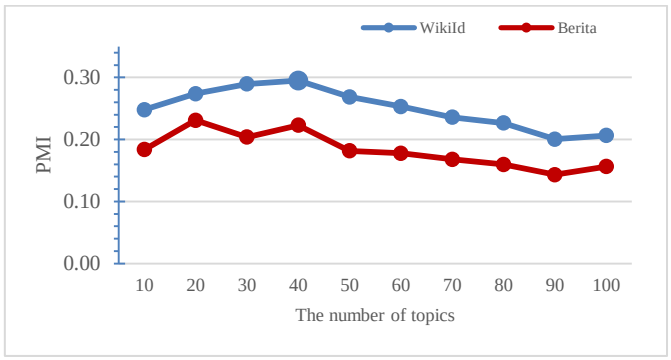


Figure 5. The Accuracy of Topics Extracted on March 12, 2016

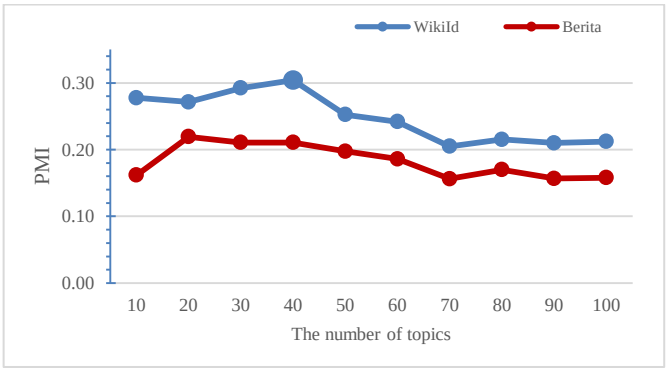


Figure 6. The accuracy of topics extracted on March 13, 2016

Figure 7 draws the trends of 40 topics extracted on March 12, 2016. From Figure 7 we can see the popular topics and their popularity changes over the period. Due to its popularity in many applications, we also visualize the trends of 100 topics extracted in the same period in Figure 8. From Figure 8, we can identify popular topics easily because the topics become more specific so that each tweet is associated to more relevant topics. For example, there is a topic presented by green curve suddenly increase at a certain period. The increasing curve implies that the topic is submitted intensively in that time but not in other time. However, the rising topics are not always to be important topics, i.e. spam tweets. Hence, we need further analysis to know what the topics are about. Moreover, the interpretability score of 100 topics is smaller than one of 40 topics as shown in Figure 5. A possible cause of this result is that the unpopular topics have low PMI scores. Therefore, if we consider only the trends of popular topics, then we may extract 100 topics to monitor them easily.

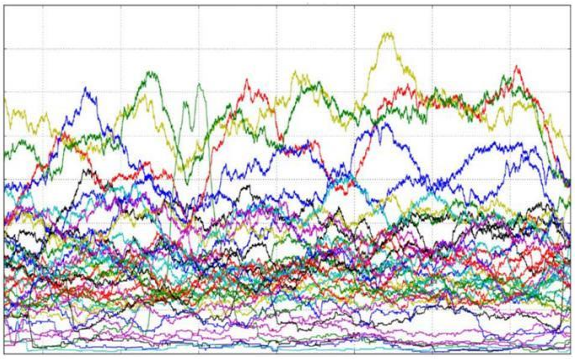


Figure 7. Trends of 40 topics extracted on March 12, 2016

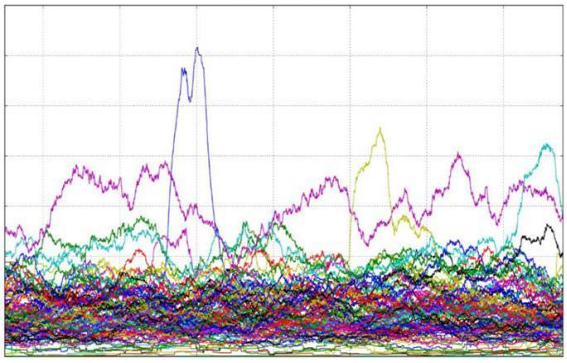


Figure 8. Trends of 100 topics extracted on March 12, 2016

We conduct similar simulations for tweets collected on March 13, 2016. Figure 9 shows the trends of 40 topics extracted on March 13, 2016, while Figure 10 gives the trends of 100 topics extracted on a similar date. Similar to the previous day, we can identify popular topics more easily in 100 topics than in 40 topics. There is also a topic that suddenly increases at a certain period. The topic has a similar trend to the topic from the previous day. After further analysis, we realize that the topic is an advertisement. These results support our earlier suggestion that we may extract 100 topics to monitor the most frequent topics easily.

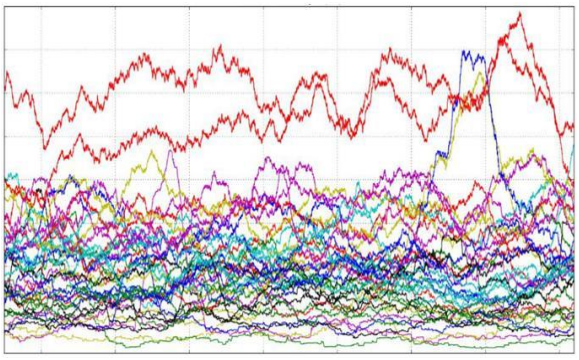


Figure 9. Trends of 40 topics extracted on March 13, 2016

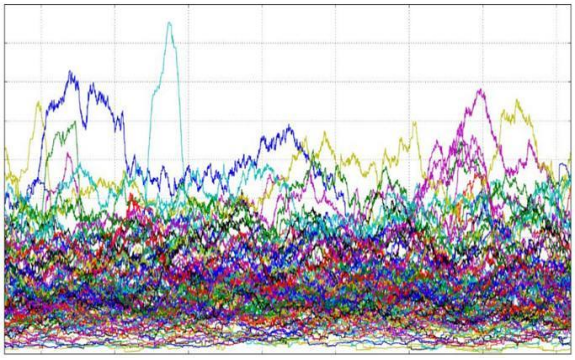


Figure 10. Trends of 100 topics extracted on March 13, 2016

4. CONCLUSION

Topic detection, such as non-negative matrix factorization, can realize Twitter as a media for urban monitoring in Greater Jakarta area. Using interpretability or coherence scores, we show that the extracted topics have similar scores to the previous simulations for other textual data. Moreover, the reference corpus

WikiId gives higher scores than the reference corpus Berita for every simulation. Therefore, WikiId may be the first candidate in calculating the coherence scores in Indonesian. In the visualization of trends, we can identify popular topics more easily in 100 topics than in the optimal number of topics. Therefore, if we consider only the trends of popular topics, then we may extract 100 topics to monitor them easily.

REFERENCES

[1] S. Goyal, "Facebook, Twitter, Google+: Social Networking", *International Journal of Social Networking and Virtual Comminities*, vol. 1, no. 1, 2012.

[2] H. Bagheri, "Big Data: Challenges, Opportunities and Cloud Based Solutions", *International Journal of Electrical and Computer Engineering*, vol. 5, no. 2, pp. 340-343, 2015.

[3] L.A. Aiello, et al. "Sensing Trending Topic in Twitter", *IEEE Transaction on Multimedia*, vol. 15, no. 6, pp. 1268-1280, 2013.

[4] D. Quercia, "Talk of the City: Our Tweets, Our Community Happiness", *Proceedings of the Sixth International AAAI Conference on Weblogs and Social Media*, pp. 965-968, 2012

[5] J. Allan, *Topic Detection and Tracking: Event-based Information Organization*. Norwell, USA, 2002.

[6] D.D. Lee and H.S. Seung, "Learning the parts of object by nonnegative matrix factorization", *Nature*, vol. 402, pp. 788-791, 1999.

[7] J.H. Lau, et al, "Machine Reading Tea Leaves: Automatically Evaluating Topic Coherence and Topic Model Quality", *14th Conferences of the European Chapter of the Association for Computational Linguistic*, pp. 530-539, 2014.

[8] R.B. Yates and B.R. Neto, "Modern Information Retrieval", *Addison Wesley Longman*, 1999

[9] M.W. Berry, *et al*., "Algorithms and applications for approximate nonnegative matrix factorization", *Computational Statistic & Data Analysis*. vol. 52, pp. 155-173, 2007.

[10] P. Paatero and U. Tapper, "Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values", *Environmetrics*, vol. 5, pp. 111–126, 1994.

[11] D.D. Lee and H.S. Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing System*. 2001; 556–562.

[12] C. J. Lin, "Projected Gradient Methods for Non-negative Matrix Factorization", *Journal Neural Computation*, vol. 19, no. 10, pp. 2756-2779, 2007.

[13] H. Kim and H. Park, "Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method", *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 2, pp. 713–730, 2008

[14] J. Kim and H. Park, "Fast nonnegative matrix factorization: an active-set-like method and comparisons", *SIAM Journal on Scientific Computing*, vol. 33, no. 6, pp. 3261-3281, 2011.

[15] C. Boutsidis and E. Gallopoulos, "SVD based initialization: a head start for nonnegative matrix factorization", *Pattern Recognition*, vol. 41, no. 4, pp. 1350-1362, 2008

[16] Pedregosa, *et al*., "Scikit-learn: Machine Learning in Python", *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.