# Power of Tags


Youyang Feng
University of Pittsburgh
yof15@pitt.edu

Ming Gao
University of Pittsburgh
mig82@pitt.edu

Ziqian Wu
University of Pittsburgh
ziw37@pitt.edu

Chenkai Zhao
University of Pittsburgh
chz97@pitt.edu



## ABSTRACT

In this paper, we aim to reproduce the work from *Learning Similarity Metrics for Event Identification in Social Media*, by Hila Becker et al. Social media is nowadays such a booming area where tons of posts, pictures, videos are posted. The tagging function that is usually embedded in these systems, however, falls behind the development of the system itself. When dealing with more than thousands of social media posts everyday, tagging manually is no longer a wise option, without even considering tagging the history archive posts. Our research involved rich context that allows us to learn and implement multi-feature similarity metrics for social media documents. We can enable search and browsing in modern search engines by automatically identifying events and their associated social media documents.



## CCS CONCEPTS

• Information system~ Information system application~Data mining~Clustering

## KEYWORDS

Event Identification, Social Media, Similarity Metric Learning


## 1 INTRODUCTION

This paper aims to cluster events by user-contributed Flickr documents from Learning Similarity Metrics for Event Identification in Social Media. We can significantly enable event browsing and search by automatically identifying these events since social media often hosts significant user-generated content. The paper we reproduced is practical and creative, which implemented some innovative and efficient algorithms, which were presented for identifying events by combining multiple context features of the document in various disciplined ways. The group members show great interest in social media clustering topics. Our members found this reproduction work would include a problem in the language of data mining and clustering, issues that we have just learned this semester. Hence, it provides a fresh example of applying these models, algorithms, analyses discussed in class. Our paper is the first step for the members toward organizing media from real-life events. Since we successfully reproduced Hila's research, we can explore complicated clustering methods, including distinguishing between event and non-event documents.

We followed the original researchers' methods because our team strives to achieve the best accuracy in our paper. During the training phase, our members implemented tf.idf in data preprocessing and single-pass incremental clustering. We followed the original researchers' methods because our team strives to achieve the best accuracy in our paper. During the training phase, our members implemented tf.idf in data preprocessing and single-pass incremental clustering. This project aims to learn a similarity metric based on the context features in social media documents. The algorithm used in this study tries to find indicators of the similarity of the documents using a weighted similarity consensus function.

## 2 RELATED WORK

We completely followed the original paper's work. The authors mentioned several related works in the original article. One related work reduced the number of total comparison pairs by using statistical properties to represent the data's subset. Our work used the average values of elements to represent the cluster. Our method is efficient because we do not need to compare each feature, and we only need to compare features with the centroid. Some works used optimization techniques or machine learning methods to learn the similarity metrics. The methods from related work would run faster than our methods since they know the number of clusters. We used single-pass classification and ensemble-based algorithms to learn the similarity matrix in our work. The advantages of our methods are clear, our method does not need the prior knowledge of cluster numbers, and the algorithms can hold the skewed data. Many articles studied topic detection and tracking event detection tasks, and most of them implemented natural language processing methods to cluster the features. However, our team combined various features to improve clustering performance.

## 3 DATASET

The dataset we used is derived from the dataset that was used in the original paper http://www.cs.columbia.edu/~hila/upcoming.tar.gz. Due to the computer limit, we shrink the size to a limited size so that the implementation, testing, and calculation is actually feasible.

The upcoming-wsdm10 dataset includes 270451 information from Flickr. The dataset consists of users ID, descriptive information of photographs(e.g., tags and tiles), geographical information(e.g., location and coordinates), automatically generated data (e.g., upload or content creation time), and event ID. The dataset contains 9517 events, with an average of 28.4 photographs per event, taken between 01/01/2006 to 12/10/2008. We used a smaller dataset derived from the original dataset because of limited memory. Our small dataset includes 15,000 information which is split from the original dataset in default order. We have 5,000 data for the training set, 5,000 for validation, and another 5,000 information for the test set. But we use 500 pieces of information for logistic regression because of the limited memory, which is more complicated and less time-effective than we encountered in implementing other algorithms.

## 4 METHOD

Below is the flowchart that demonstrates the processing of the model. We build up the system with reference to it.

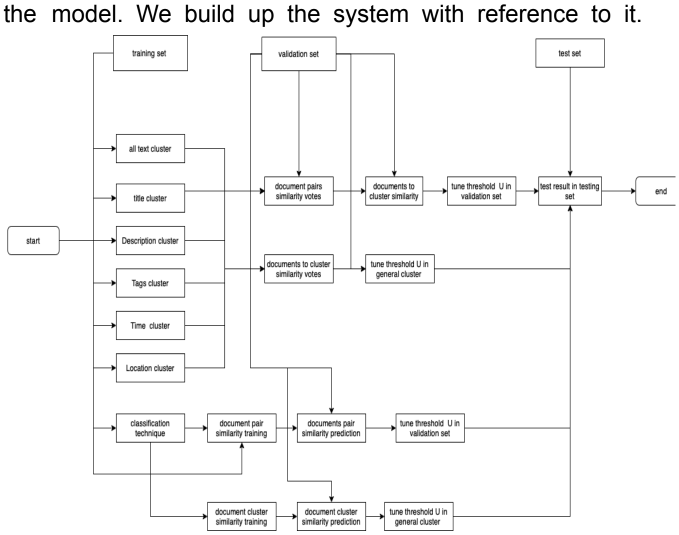


**Figure1: The flowchart of clustering process**

Before analysis, we first pre-processed the data to similarity matrices so that we could apply clustering more effectively. We represent textual features as tf.idf weight vectors and use the cosine similarity metric to evaluate their closeness. We calculate the similarity of time and date by $c = 1 - |t_1 - t_2|/y$.

The similarity of location is calculated using Haversine distance, denoted by H, as $c = 1 - H(l_1, l_2)$.

We then apply the self-implemented single pass incremental clustering algorithm to the centroid of documents, which considers each element in turn and determines the suitable cluster assignments based on the element's similarity to any existing clusters.

For the ensemble model, we select clusters that partition the data using the different features and the similarity matrices we implemented. We use the single-pass incremental clustering algorithm as the clustering similarity function.

Moreover, we select the best threshold based on each cluster's performance according to NMI and B-cubed scores. The weights are assigned during the supervised training phase, which determines the influence of each cluster *C*.

When combining individual partitions, we use a weighted binary vote for each pair of documents and clusters. When combining individual similarities, we compute the similarity between a document $d_i$ and a cluster centroid $c_j$. $Pc(d_i, d_j) = 1$if $\boldsymbol{\sigma}_c(d_i, c_j) > \boldsymbol{\mu}_c$ , and 0 otherwise. Eventually, we can get the similarity metric, which is of the form $\boldsymbol{\Sigma} cP_c(d_i, c_j)w_c$.

Next, we use classification models to learn about document similarity functions for social media. Given a pair of documents $d_i$ and $d_j$, we compute the raw similarity scores $\boldsymbol{\sigma}$ corresponding to features and similarity matrices we implemented in the preprocessing.

Innovatively, we applied word2vec to replace the tf-idf method in data preprocessing, which supposedly would generate a more robust result. Moreover, we use multilayer perceptrons in the training process because of its swiftness and nice fitting with large scale input data.

Our implementation mainly specializes in efficiency and accuracy. To improve the efficiency of the incoming training phase, we also use stop-word elimination and stemming so that the model can better analyze the text features. We also apply the centroids to represent each cluster in the single pass incremental clustering algorithm. What is more, instead of computing the consensus score using the clusters' predictions, we compute the documents' feature-specific similarity metrics directly for documents and cluster centroids. To improve accuracy, we apply NMI and

B-Cubed scores to balance our desired clustering properties, maximizing the homogeneity of events within each cluster and minimizing the number of clusters that documents for each event that are spread across.

## 5 EVALUATION RESULTS

We have two standards to evaluate the performance of our model, Normalized Mutual Information (NMI) and B-Cubed score.

NMI measures how much information is shared between actual "ground truth" events, each of which has an associated document set, and the clustering assignment.

$$NMI(C,E) = \frac{I(C,E)}{\frac{H(C)+H(E)}{2}}$$

B-cubed score estimates the precision and recall associated with each document in the dataset individually.

$$B - Cubed = 2\frac{P_b \cdot R_b}{P_b + R_b}$$

| Algorithms | NMI | B-Cubed | Threshold for best performance |
|---|---|---|---|
| Title | 0.6072 | 0.5166 | 0.3 |
| Description | 0.5014 | 0.5102 | 0.95 |
| Tags | 0.9625 | 0.9197 | 0.25 |
| Time | 0.3311 | 0.0953 | 0.95 |
| Location | 0.6935 | 0.5508 | 0.95 |

**Table 1: Performance of each feature using single-pass incremental clustering technique.**

| | Original data on paper | | Data we get from our replication | |
|---|---|---|---|---|
| Algorithms | NMI | B-Cubed | NMI | B-Cubed |
| Tags | 0.9229 | 0.7676 | 0.9625 | 0.9197 |
| ENS-PART | 0.9296 | 0.7819 | 0.9462 | 0.8933 |
| ENS-SIM | 0.9322 | 0.7861 | 0.8971 | 0.8055 |
| CLASS-LR | 0.9444 | 0.8155 | 0.6083 | 0.3810 |

**Table 2: Comparison of the performance of all similarity metric learning techniques and the best individual clustering techniques.**

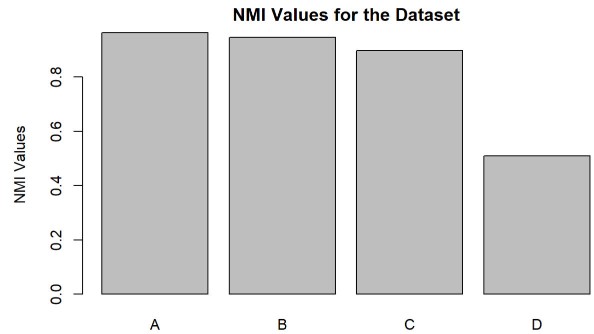


**Figure 2: NMI scores on the Upcoming dataset for Tags(A), ENS-PART(B), ENS-SIM(C), CLASS-LR(D)**

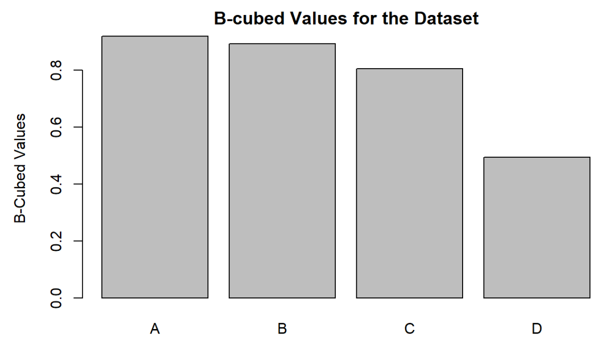


**Figure 3: B-Cubed scores on the Upcoming dataset for Tags(A), ENS-PART(B), ENS-SIM(C), CLASS-LR(D)**

## 6 DISCUSSION

As shown above, the evaluation results until now are much better than the data in the presentation last week and the performance of some clusters already exceeds the clusters given in the original paper. This is because we further optimize the core algorithms and data-processing method. However, some clusters are still unsatisfactory, such as classification-based similarity, but we believe most of the results are still within an expected and acceptable range. We figured out several reasons to explain why some models behave worse than original results.

The first reason is that we applied a smaller dataset, especially for classification-based similarity because the computation will increase dramatically if multiplying the amount of data. But the smaller size indicates that less data are included, so the model trained from this dataset could perform differently compared to the original one.

The second reason is the underlying randomness of the implementation. The sampling that picks training, testing, and validating dataset is de-facto random, which means the result can vary if we apply a different grouping strategy.

Our innovation may also lead to this different result. Recall we use word2vec instead of tf-idf method in data preprocessing. Although technically this replacement should bring a cleaner and more reliable dataset, since it is different from the original tf-idf method, which is the base of all the following steps in the original paper, it is possible that this innovation may require other adjustments to the following steps to perfectly fit this model.

Lastly, the implementation detail can be different. Since the original paper does not contain code for reference, we implemented the single pass incremental clustering ourselves. This algorithm is the core of the whole system. Suppose we had slightly different implementations in some steps, the result could be totally different. This reason is totally a reasonable hypothesis. Without checking the original implementation, it is impossible to testify the correctness of this guess.

## 7 CONCLUSION

This paper has met most of our expectations. It produced decent results from limited resources. Although we have to admit that the model is not a perfection yet, the implementation itself is already a success. Moreover, the ultimate goal of this paper is not to completely simulate the exact same result that the original paper presents, but rather to prove the feasibility of the model and algorithm. With all these standards considered, we believe this project is, after all, a notable achievement.

## 8 WORK ASSIGNMENT

We divide our project, in terms of algorithm, into four parts, where each of our members is responsible for one subcomponent. Chenkai Zhao is responsible for general single-pass incremental clustering algorithm implementation, Ziqian Wu for converting document pairs similarity to all-feature clusters, Ming Gao for calculating document and feature-based cluster similarity to generate the all-feature cluster, and Youyang for classification implementation to get document similarity. For the paperwork the presentation slides, we worked together in a collective manner.

## REFERENCES

[1] Becker, H., 2010. *Learning Similarity Metrics for Event Identification in Social Media*. [online] http://www.cs.columbia.edu/~hila/papers/wsdm10-becker.pdf. Available at: <http://www.cs.columbia.edu/~hila/papers/wsdm10-becker.pdf> [Accessed 13 December 2021].