Friendship Paradox and Hashtag Embedding in the Instagram Social Network

Conference Paper · October 2019

DOI: 10.1007/978-3-030-33110-8_11

3 authors, including:

Miroslav Mirchev
Ss. Cyril and Methodius University in Skopje
37 PUBLICATIONS **172** CITATIONS
SEE PROFILE

Igor Mishkovski
Ss. Cyril and Methodius University in Skopje
68 PUBLICATIONS **361** CITATIONS
SEE PROFILE

Friendship paradox and hashtag embedding in the Instagram social network

David Serafimov, Miroslav Mirchev and Igor Mishkovski

Faculty of Computer Science andd Engineering,
Ss. Cyril and Methodius University in Skopje, North Macedonia

Abstract. Instagram is a social networking platform which gained popularity even faster than most of the other modern online social networks. It is relatively newer and less explored than other social networks, such as Facebook and Twitter. Therefore, we have conducted a research based on a sample data set extracted through the Instagram weekend hashtag project, in order to unveil some of its characteristics. First, we reveal the various forms of friendship paradox present in Instagram, which are often observed in social networks. Then, we conduct a detailed hashtag analysis and provide a method for hashtag representation and recommendation using natural language processing.

Keywords: online social networks, network science, natural language processing

1 Introduction

Online social networks (OSNs) have been widely studied in the past [1–3], however, Instagram is relatively newer and less researched. Knowledge discovery from this network can help gain a deeper insight into the processes that drive its growth, as well as reveal some characteristics of other social networks in the real world for which there is no available data. A thorough study of Instagram was presented in [4] where users and photos were divided into several categories based on network and photographic data. In [5] the authors provide a detailed analysis of the liking activity among the Instagram users. The behavior of the silent users, known as lurkers, was explored in [6] for several OSNs including Instagram. In this paper we focus on two topics, revealing the existence of the friendship paradox and providing a suitable hashtag embedding.

The friendship paradox is a phenomenon discovered in 1991 [7] and it states that "most people have fewer friends than their friends have, on average". On the contrary, usually people think that they have more friends than most of their friends. This phenomenon is not limited to friends and can be observed in social networks with other types of relationships. An example of this is the social network of partners. Most of the individuals in this network have fewer partners than their partners on average. The friendship paradox can be also applied in predicting epidemic spreading as well as immunization [8]. In addition to the real world, this paradox is also present in the online world. One example is the social network Twitter [9]. In this social network, more than 98% of users had fewer followers than their followers, on average. The friendship paradox have been explored in other social networks such as Facebook [10], but to the best of our knowledge it has not been confirmed for Instagram. Here we will show the friendship paradox using a dataset extracted from Instagram, which was already explored in [11] for studying other relevant interesting aspects.

We will use the same data set to provide a hashtag analysis in Instagram using natural language processing, which to the best of our knowledge have not be done elsewhere. Several approaches have been published to get multidimensional representations of hashtags. These methods depend on additional features like images [12], text [13, 14], or some other. The nature of these methods does not allow us to easily adapt them for data sets where such features are not available. On the other hand, in our study we rely only on the available hashtags. These network analyses could be useful in exploring the spread of trends across the network [15], and potentially their control and timely prevention.

The paper is organized in the following way. In Section 2 we describe the Instagram data set used in our study. In Section 3 we explain the friendship paradox and present our analysis for Instagram. In Section 4 we show a method for hashtag representation and based on it two models for hashtag recommendation. We finalize the paper with some conclusions in Section 5.

2 Data set

In this paper we study the social network Instagram through a data set based on the weekend hashtag project, which was first used and described in [11]. The weekend hashtag project is a competition that is held every Friday and is organized by the Instagram team. The contest consists of a unique hashtag with the #whp prefix, which is also the theme of the contest. Users of the social network can participate in the competition if they post a picture with the designated hashtag. To collect this data set, 72 WHP hashtags were selected. 2081 users who joined one of these 72 competitions, were randomly selected. Data about what these users shared was also collected. A breadth first search is started from the seed users, skipping any users who did not participate in the contests.

The data set consists of two files. The first file in each row contains: follower id, followed user id, number of likes from the follower to the followed, number of comments from the follower to the followed, and timestamps for all the comments. The second file consists of data about what users posted, where each row contains: posted picture id, number of shares for the post, post timestamp, hashtags included in the post, the number of likes for the post, and the number of comments for the picture. In the data set we have a total of $1,686,349$ posts. These posts were posted by $2,081$ different users. A total of $8,919,630$ hashtags were included in all posts, from which $269,359$ were unique. The total number of likes was $1,242,923,022$ and the number of comments was $41,341,783$. There are a total number of $44,766$ users. There are $677,686$ connections between these users. The average node degree in the network is 15.14. The average length of the shortest path in this network is 3.16, although the longest shortest path is 11. The network has 151 communities. The clustering coefficient is 0.041, the assortativity coefficient is -0.097, and the modularity of the network is 0.578.

3 Friendship Paradox in Instagram

In the Instagram social network, users can do several activities such as follow users, post images or videos, like, comment, etc. In this section we examine whether the friendship paradox applies to this social network and for which activities. The relationships in Instagram are directed, so if we follow someone, they do not have to follow us. The people that follow us are our followers and those that we follow are our followees. Therefore, we check if a friendship paradox occurs both relating to followers and followees in Instagram. This kind of directed relationships are similar to Twitter for which the various forms of friendship paradox have been explored in [9]. On the other hand, in Facebook the relationships are mutual and the graph is undirected, allowing the application of the classical friendship paradox [10].

The friendship paradox can be rephrased in the two following ways:

(i) Our *followees* have or do something more than us on average.
(ii) Our *followers* have or do something more than us on average.

We also check the presence of both weak and strong friendship paradox, where the *weak* paradox is calculated compared to the average and the *strong* is compared to the median in order to reduce the impact of the extremes. In the following text we present the calculated friendship paradox for different activities and the results are summarized in Fig. 1 for the paradox relating to the followees and in Fig. 2 for the paradox relating to the followers. In the text we will mostly comment the results for the weak paradox, while the results for the strong paradox usually follow a similar pattern and the reader can see them in the figures. We only comment the strong paradox in the number of hashtags, because it does not always apply there.

Followers Following a user in Instagram allows you to see what they post. By analyzing the data we confirmed a friendship paradox for the number of followers that both our followees and followers have. On average 91.71% of the users are followed by fewer users than their followees, while 73.64% of the users have less followers than their followers.

Followees The friendship paradox was also observed in the number of followees, so the majority of users follow less people than their followers and followees. On average 82.21% of the users follow less users than their followees, and 76.99% of the users have less followees than their followers.

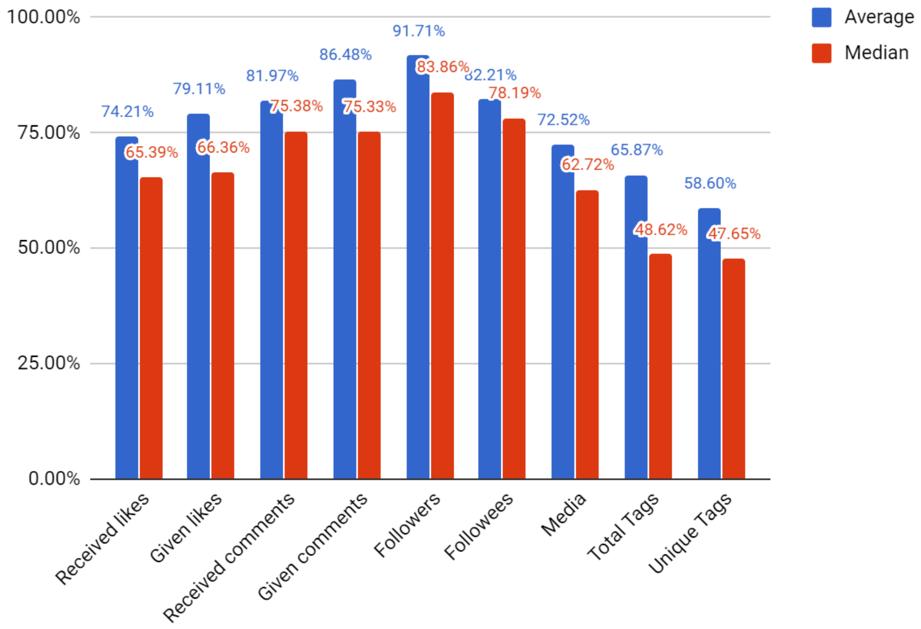

Fig. 1. User percentage for which the friendship paradox relating to the followees applies for various properties.

Likes Users in Instagram can share likes on posts or comments, and thus let other users know that they like something and the friendship paradox was also shown for the number of likes. On average 74.21% of the users have received less total likes than their followees, while 71.44% of the users have received fewer likes than their followers. The paradox of friendship is also true in the number of likes given. On average 79.11% of users have given less likes than their followees, and 77.57% than their followers.

Comments In addition to likes, Instagram users can also share text as comments on posts or other activities. With the help of comments, users can share their opinions, discuss publicly on a topic, etc. On average 86.48% of the users have given fewer comments than their followees, while 76.78% have posted less comments from their followers. Similarly, 81.97% of the users have received less comments on their posts than their followees on average, and 69.86% than their followers.

Posts and hashtags Instagram users can post and share media. When sharing, users often add hashtags. A hashtag is a string of characters that starts with a hash ('#'). There are several benefits to adding hashtags, such as giving context to a shared image or easier discovery of relevant content. On average 72.52%

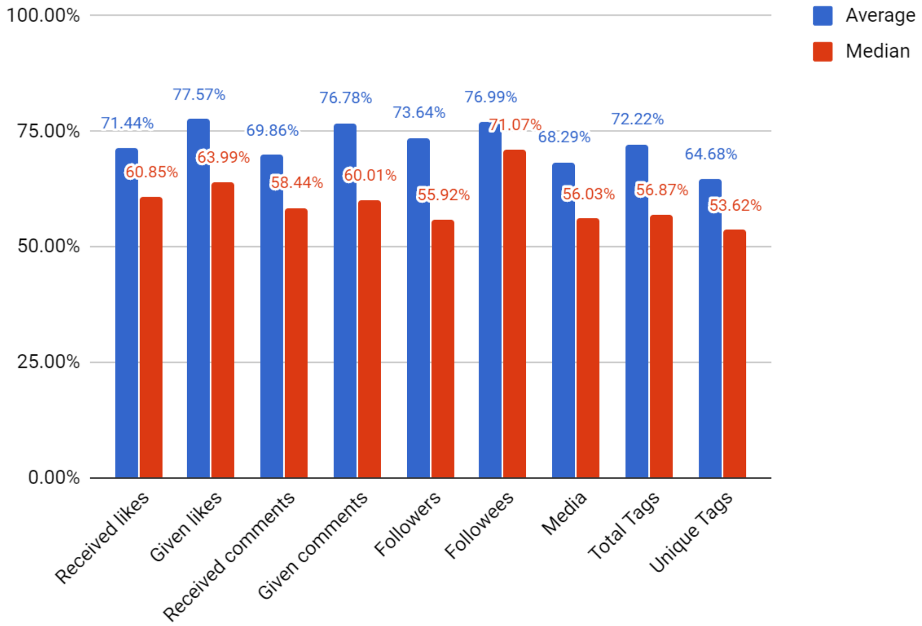

Fig. 2. User percentage for which the friendship paradox relating to the followers applies for various properties.

of the users share less images and videos than their followees, and 68.29% than their followers. In addition, on average 65.87% of the users used less hashtags per photo than their followees, and 72.22% than their followers. If we consider only unique hashtags, then on average 58.60% of the users have used less unique hashtags per photo than their followees on average, and 64.68% than their followers.

We can conclude that in Instagram's social network, as in other online social networks, we can observe a friendship paradox for various network activities. The weak variant of the friendship paradox applies to all network criteria, while the strong variant of the friendship paradox applies to all criteria except for the number of total and unique hashtags used compared to our followees. For most of the activities the number of followees who have or do something more than us is higher than the number of followers, which is expected, except for the number of hashtags. A higher number of our followers have both total and unique hashtags than our followees. This analysis can help us to better evaluate the impressions and behavior of Instagram users and understand the spreading trends in the social network.

4 Hashtag analysis with natural language processing

Besides examining the friendship paradox we also conducted an analysis of the hashtags used in Instagram using natural language processing (NLP). NLP is a discipline concerned with the understanding of natural languages and their representation in machines. For example, the words ”dog” and ”puppy” have similar semantic meaning, but are composed from different letters and have different lengths. One of the problems that is explored in NLP is how to represent words in such a way that we preserve these semantic meanings, without depending on human intervention.

A popular solution to this problem is the term frequency-inverse document frequency method [16]. For this method we need a corpus of N documents. We represent a word as an N dimensional vector, where the i-th value represents how important the word is to the i-th document based on the occurrence of the word in the document and in the corpus. Let us say the first three documents in the corpus deal with biology. Then words related to biology would have higher values in the first three dimensions, and lower values in the other dimensions. If we calculate the angle between the direction of the vectors, we will get smaller value for words that appear in similar documents, and larger for words that appear in documents about different topics. This method of word presentation has some drawbacks. Vectors are sparse and long. The quality of the vectors depends greatly on the corpus, and it is challenging to preserve subtle differences between similar words.

4.1 Word2Vec

A more sophisticated method for solving this problem is using Word2Vec vectors [17]. This method works using neural networks and a corpus of documents. There are two main approaches. The first approach is to train a neural network to predict words based on its surroundings. The second approach is the opposite, training a neural network to predict the surroundings based on a given word. Both approaches have similar performance. The first approach is generally faster, while the second approach is slower, but the resulting vectors are of higher quality for words that rarely appear in the training corpus. Words that appear in similar environments are represented by vectors with lower difference between their angles. Words that appear in different environments are represented by vectors with greater angle difference between them. The obtained vectors are often used as input for other NLP models, but can be used on their own for solving simpler problems. Other similar methods for word representation in multidimensional space were discovered later, but the obtained vectors did not significantly improve [18, 19].

4.2 Hashtag representation

Although hashtags are different from words, they share many properties. Many of the problems that we encounter in NLP also exist for hashtags, for example:

Table 1. Bag of hashtags example

Input	Output
#mycat #cats #catsofig	#cat
#cat #cats #catsofig	#mycat
#cat #mycat #catsofig	#cats
#cat #mycat #cats	#catsofig

named entity recognition, sentiment analysis, etc. Another common problem is how to represent hashtags. Although several approaches exist, often they depend on additional information like images [12] or text [13, 14]. In this paper we will describe a method for embedding hashtags in a multidimensional space based on the Word2Vec method. The input of this method are hashtags that have appeared together. Since we do not have any additional constraints, the proposed method is more general and platform agnostic.

4.3 Architecture

The method consists of a neural network with one hidden layer. On the input we have one hot encoding, meaning that each input node is one hashtag. The hidden layer will have as many nodes as the dimensions of the resulting hashtag vectors. On the output layer we also have one hot encoding. As in Word2Vec, we have two approaches and we will show the difference between them using the following set of hashtags {#cat, #mycat, #cats, #catsofig}. In the first approach the input is all hashtags but one, and the output is the left out hashtag. From one sample with N hashtags we get N inputs and outputs. All the inputs and outputs of the example can be found in Table 1. This approach will be reffered to as ”bag of hashtags” in this paper.

In the second approach the input is one hashtag, and the output is one of the surrounding hashtags. From one sample with N hashtags we get $N \times (N - 1)$ input output pairs. All the inputs and outputs for the sample can be found in Table 2. This approach will be refereed to as ”hashtag pairs” in this paper.

There are several hyperparameters in both approaches:

Number of nodes in the hidden layer. The number of nodes in this layer is equal to the dimensions of the vectors that are obtained at the end. In general, more dimensions can store more information about the hashtags.

Epochs of training. The number of epochs corresponds to the number of times each sample will be used for training the model. The number of epochs linearly increases the training time of the model and improves the quality of the resulting vectors.

Minimum number of occurrences of one hashtag. The number of times that a hashtag should appear in the data set to be included in the vocabulary.

Table 2. Hashtag pairs example

Input	Output
#cat	#mycat
#cat	#cats
#cat	#catsofig
#mycat	#cat
#mycat	#cats
#mycat	#catsofig
#cats	#cat
#cats	#mycat
#cats	#catsofig
#catsofig	#cat
#catsofig	#mycat
#catsofig	#cats

Table 3. Evaluation input-output pairs

Input	Output (recommendation)
#mycat #cats #catsofig	#cat
#cat #mycat #cats	#catsofig

4.4 Evaluation

For evaluating the vector quality, we will use the Instagram data set. Input-output pairs are not available, so we either have to try unsupervised learning or generate a set of input-output pairs. Since we want to quantify the quality of the vectors, we choose the second approach. The data set does not provide good assumptions about additional hashtags that we could recommend, so instead we have to use the existing data to generate recommendations. We assume that if a user has posted a certain hashtag in a post with multiple hashtags, the hashtag is a good recommendation for the remaining hashtags. This is how evaluation looks like for a single post:

Let us assume the user posted a picture with the following hashtags: #cat, #mycat, #cats and #catsofig. Some hashtags are removed for evaluation, for example #cat and #catsofig. During training we only have #mycat and #cats which we use to generate hashtag embedding. During evaluation we will have the input-output pairs given in Table 3. Such samples, although not perfect, should approximate good recommendations.

4.5 Experiment

The experiment consists of training and evaluating six models of the presented architecture with different hyperparameters compared with a baseline model.

Table 4. Pairs of hashtags and their similarity calculated with cosine distance

First hashtag	Second hashtag	Similarity
#instagood	#instamood	0.93206483
#christmas	#xmas	0.87076986
#rap	#rnb	0.74523616
#dad	#father	0.74124840
#netflix	#cats	0.24433972
#nofilter	#sanfrancisco	0.17591012
#instagood	#garden	0.10391730

The baseline model is a simple statistical model that gives recommendations according to previous occurrences of different hashtags. The data set is split into 90% training and 10% test sets. All models were trained for 50 epochs. Hashtags in the vocabulary have occurred at least 3 times in the data set. Three of the models were trained with the bag of hashtags method, and three with the hashtag pairs method. Two models were trained with 64 nodes in the hidden layer, two with 128 nodes in the hidden layer, and two with 256 nodes in the hidden layer. Learning was executed on the Intel Xeon Scalable processor with 3.7 Gigabytes RAM hosted on the Google Cloud Platform. Model learning takes between one and four hours depending on the method and the hyperparameters on the described hardware. The source code is available on github [20]. We will use the recall at K (R@K) metric to measure quality, which is calculated as the average relevant hashtags that are recommended in the top K, for $K \in 1, 2, 3, 5, 10$.

4.6 Usage

With the resulting vectors, we can perform some operations that were previously difficult or not possible. We can search for similar hashtags, calculate the similarity between hashtags, group hashtags by topics, and even do arithmetic operations with hashtags. All examples in the thesis were obtained from a hashtag pairs model with 64 hidden nodes.

Calculating hashtag similarity Several metrics exist for calculating the distance between two vectors, for example: Euclidean distance, Manhattan distance, cosine distance, etc. If we represent two hashtags as vectors, then with one of the above metrics we could calculate the distance or similarity between the hashtags. Vectors are derived from weights in the neural network, so cosine distance makes most sense in this case. Table 4 provides some examples.

Searching similar hashtags If for a hashtag we calculate the cosine distance with all other hashtags for which we have calculated vectors, we can find the most similar hashtags. Table 5 gives a few examples.

Table 5. Hashtags and their closest neighbors according to cosine distance

Data set	Target hashtag			
	#christmas	#vsco	#istanbul	#healthy
1	#christmastree	#vscocam	#igersistanbul	#eatclean
2	#xmas	#vscophile	#feelingistanbul	#cleaneating
3	#santa	#vscofeature	#turkishfollowers	#igfit
4	#ornaments	#vscogram	#turkey	#exercise
5	#carols	#afterlight	#ig_turkey	#eatingclean
6	#christmaslights	#newvscocam	#hayatandanibarettir	#nutrition
7	#presents	#vscofilm	#igersturkey	#dialabreakfast

Clustering hashtags The obtained vectors can be clustered and Fig. 3 shows hierarchical clustering using cosine distance metric and average of clusters as linkage criteria.

Arithmetic operations We can perform some arithmetic operations, such as addition and subtraction, with the obtained vectors. If we search for the nearest hashtag of the resultant vector, we can find an approximation to the result. Here are some examples:

#helloween − #pumkin + #christmas = #christmastree
#kids − #little_igers + #cats = #catstagram
#sweden − #stockholm + #turkey = #istanbul
#woods − #forest + #city = #buildings

Calculating post distance based on included hashtags Another interesting feature of word vectors is that with their help we can calculate the distance between documents. Word Mover's Distance [21] is a technique based on Earth Mover's Distance with which we can calculate the similarity of two documents based on the similarity of the words appearing in each document. If we adapt this technique to hashtag embedding, we can calculate the similarity between two images.

4.7 Results

A simple way to find recommendations for n hashtags, is to find the closest hashtags to their average. Although this method is naive, the results show that in practice it is much better than our baseline statistical model. In Table 6 we can see the results. The best results for each metric are in bold.

We can observe that the hashtag pairs method generally performs better on the given task. The number of dimensions also impacts the outcome and the highest dimensional models did not performed better on this task. A possible

Table 6. The recall at K (R@K) metric for the bag of hashtags (BoH), hashtag pairs (HP), with 64, 128 and 256 dimensions (D), as well as the baseline statistical (BS) model.

Model	R@1	R@2	R@3	R@5	R@10
BS	0.0201	0.0366	0.0480	0.0703	0.1184
64D BoH	0.0617	0.0865	0.1035	0.1274	0.1661
64D HP	0.0779	0.1157	0.1435	**0.1836**	**0.2414**
128D BoH	0.0339	0.0477	0.0572	0.0713	0.0956
128D HP	**0.0824**	**0.1189**	**0.1445**	0.1801	0.2307
256D BoH	0.0358	0.0504	0.0623	0.0825	0.1295
256D HP	0.0769	0.1091	0.1296	0.1573	0.1942

explanation is that the training set is too small to take advantage of the additional dimensions and the models became overfitted. All models perform on par or better than the baseline model for all metrics. The 128-dimensional model obtained with the hashtag pairs method has the best results for the metrics R@1, R@2 and R@3. The smaller 64-dimensional model performed best for R@5 and R@10 metrics.

5 Conclusion

In this paper we provided some network and hashtag analysis of the Instagram network using a sample data set. In the first part we confirmed both the strong and weak variant of the friendship paradox in the network for many network properties, such as the number of followers, likes, posts, hashtags and comments, both regarding the followers and the followees. Solely for the number of total and unique hashtags compared to the followees, only a weak paradox was observed. Generally the friendship paradox is stronger for the followees than for the followers, except for the number of hashtags used.

We also introduced a general method for obtaining high-quality hashtag representations in multidimensional space. We proposed a method for obtaining a data set for the task of hashtag recommendation. We have tested the obtained hashtag embedding on the given problem and the results showed improvement compared to the baseline model. It is fair to assume that vectors obtained with the proposed models will contribute to improving models that depend on high-quality hashtag representations, similarly as word representations have improved models that depend on them.

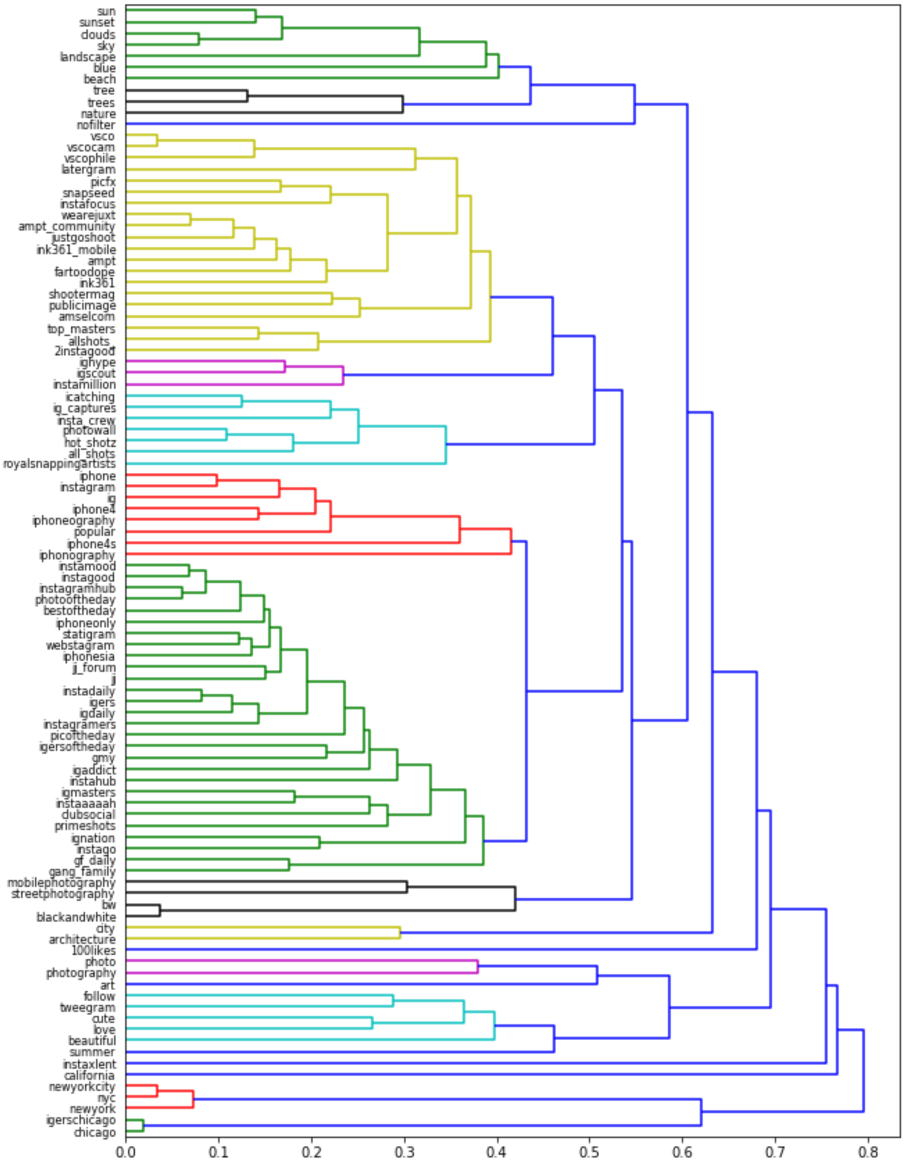

Fig. 3. Hierarchical clustering of the 100 most common hashtags in the dataset.

References

1. Alan Mislove, Massimiliano Marcon, Krishna P Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 29–42. ACM, 2007.
2. Ravi Kumar, Jasmine Novak, and Andrew Tomkins. Structure and evolution of online social networks. In *Link mining: models, algorithms, and applications*, pages 337–357. Springer, 2010.
3. Janice Penni. The future of online social networks (osn): A measurement analysis using social media tools and application. *Telematics and Informatics*, 34(5):498–517, 2017.
4. Yuheng Hu, Lydia Manikonda, and Subbarao Kambhampati. What we instagram: A first analysis of instagram photo content and user types. In *Eighth International AAAI conference on weblogs and social media*, 2014.
5. Jin Yea Jang, Kyungsik Han, and Dongwon Lee. No reciprocity in liking photos: analyzing like activities in instagram. In *Proceedings of the 26th ACM Conference on Hypertext & Social Media*, pages 273–282. ACM, 2015.
6. Andrea Tagarelli and Roberto Interdonato. Time-aware analysis and ranking of lurkers in social networks. *Social Network Analysis and Mining*, 5(1):46, 2015.
7. Scott L Feld. Why your friends have more friends than you do. *American Journal of Sociology*, 96(6):1464–1477, 1991.
8. Reuven Cohen, Shlomo Havlin, and Daniel Ben-Avraham. Efficient immunization strategies for computer networks and populations. *Physical review letters*, 91(24):247901, 2003.
9. Nathan O Hodas, Farshad Kooti, and Kristina Lerman. Friendship paradox redux: Your friends are more interesting than you. In *Seventh International AAAI Conference on Weblogs and Social Media*, 2013.
10. Keith N Hampton, Lauren Sessions Goulet, Cameron Marlow, and Lee Rainie. Why most facebook users get more than they give. *Pew Internet & American Life Project*, 3:1–40, 2012.
11. Emilio Ferrara, Roberto Interdonato, and Andrea Tagarelli. Online popularity and topical interests through the lens of instagram. In *Proceedings of the 25th ACM conference on Hypertext and social media*, pages 24–34. ACM, 2014.
12. Andreas Veit, Maximilian Nickel, Serge Belongie, and Laurens van der Maaten. Separating self-expression and visual content in hashtag supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5919–5927, 2018.
13. Jason Weston, Sumit Chopra, and Keith Adams. # tagspace: Semantic embeddings from hashtags. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1822–1827, 2014.
14. Bhuwan Dhingra, Zhong Zhou, Dylan Fitzpatrick, Michael Muehl, and William W Cohen. Tweet2vec: Character-based distributed representations for social media. *arXiv preprint arXiv:1605.03481*, 2016.
15. Leihan Zhang, Jichang Zhao, and Ke Xu. Who creates trends in online social media: The crowd or opinion leaders? *Journal of Computer-Mediated Communication*, 21(1):1–16, 2015.
16. Anand Rajaraman and Jeffrey David Ullman. *Mining of massive datasets.* Cambridge University Press, 2011.

17. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
18. Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.
19. Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.
20. https://github.com/nasadigital/diplomska-instagram.
21. Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *International Conference on Machine Learning*, pages 957–966, 2015.