DOI: 10.15514/ISPRAS-2019-31(5)-10

Methods for News Items Popularity Estimation on Early Stages

[2] A.A. Avetisyan, ORCID: 0000-0002-7066-6954 <a.a.avetisyan@ispras.ru>
[1] M.D. Drobyshevskiy, ORCID: 0000-0002-1639-9154 <drobyshevsky@ispras.ru>
[1,2] D.Yu. Turdakov, ORCID: 0000-0001-8745-0984 <turdakov@ispras.ru>

[1] Ivannikov Institute for System Programming of the RAS,
25, Alexander Solzhenitsyn Str., Moscow, 109004, Russia
[2] Lomonosov Moscow State University,
GSP-1, Leninskie Gory, Moscow, 119991, Russian Federation

Abstract. Millions of news are distributed online every day. Tools for predicting the popularity of news stories are useful to ordinary people to discover important information before it becomes generally known. Also, such methods can be used to increase the effectiveness of advertising campaigns or to prevent the spread of fake news. One of the important features for predicting information spread is the structure of the influence graph. However, this feature is usually not available for news, because authors rarely post explicit links to information sources. We propose a method for predicting the most popular news in the information flow, which solves this problem by constructing a latent graph of influence. Computational experiments with two different datasets have confirmed that our model improves the precision and recall of forecasting the popularity of news stories.

Keywords: information diffusion; information cascades; networks of diffusion

For citation: Avetisyan A.A., Drobyshevskiy M.D., Turdakov D.Yu. Methods for News Items Popularity Estimation on Early Stages. Trudy ISP RAN/Proc. ISP RAS, 2019, vol. 31, issue 5, pp. 137–144. DOI: 10.15514/ISPRAS-2019-31(5)-10

Acknowledgements. The study was funded by RFBR according to research project 18-07-01059.

Методы оценки популярности новостных материалов на ранних стадиях

[2] А.А. Аветисян, ORCID: 0000-0002-7066-6954 <a.a.avetisyan@ispras.ru>
[1] М.Д. Дробышевский, ORCID: 0000-0002-1639-9154 <drobyshevsky@ispras.ru>
[1,2] Д.Ю. Турдаков, ORCID: 0000-0001-8745-0984 <turdakov@ispras.ru>

[1] Институт системного программирования им. В.П. Иванникова РАН,
109004, Россия, г. Москва, ул. А. Солженицына, д. 25
[2] Московский государственный университет имени М.В. Ломоносова,
119991, Россия, Москва, Ленинские горы, д. 1

Аннотация. Миллионы новостей распространяются онлайн каждый день. Инструменты для прогнозирования популярности новостных материалов полезны для простых людей, чтобы обнаружить важную информацию, прежде чем она станет общеизвестной. Также такие методы можно использовать для повышения эффективности рекламных кампаний или предотвращения распространения поддельных новостей. Одной из важных особенностей прогнозирования распространения информации является структура графа влияния. Однако обычно для новостей она неизвестна, поскольку авторы редко публикуют явные ссылки на источники информации. Мы

предлагаем метод прогнозирования наиболее популярных новостей в информационном потоке, который решает эту проблему путем построения скрытого графа влияния. Вычислительные эксперименты с двумя различными наборами данных подтвердили, что наша модель повышает точность и точность прогнозирования популярности новостных сообщений.

Ключевые слова: распространение информации; информационные каскады; сети распространения

Для цитирования: Аветисян А.А., Дробышевский М.Д., Турдаков Д.Ю. Методы оценки популярности новостных материалов на ранних стадиях. Труды ИСП РАН, том 31, вып. 5, 2019 г., стр. 137-144 (на английском языке). DOI: 10.15514/ISPRAS-2019-31(5)-10

Благодарности. Исследование финансировалось РФФИ в соответствии с исследовательским проектом 18-07-01059.

1. Introduction

Prediction of the popularity of data streams can be used in various scenarios such as political campaigns, preventing the spread of fake news, viral advertising, etc. In recent works, four types of features are distinguished for information popularity prediction: temporal features, structural features, features of early adopters [1], and content features. Structural features are based on explicit graphs, therefore they are applied only for analysis of online social networks.

Most of news platforms do not have explicit links between each other. As a result, it becomes impossible to track how the interaction between different resources affects the popularity of the news. However, we can assume the existence of a hidden network of influence between news sources and try to reconstruct it.

Thereby, we emphasise the two directions of studying information propagation: 1) inferring an influence graph under the assumption that information is spread along its edges, and 2) prediction of news popularity based on information flow features.

We define information propagation for a particular message as a chain

$$c = \big((u_1;\, t_1;\, \Theta_1);\, (u_2;\, t_2;\, \Theta_2), \ldots, (u_n;\, t_n;\, \Theta_n)\big),$$

where u_i is the i-th disseminator who posted the message, t_i is a corresponding publication time, and Θ_i is an information about the post. We will call such chains *cascades*. A cascade will be considered popular if it contains more messages than n% of other cascades in the flow. The early stage of a cascade is the time when a small fixed number k of disseminators posted the message. In this paper we solve the following problem: to predict at the early stage with $k = 5$ whether a cascade will become larger than 50% of the other cascades in the flow.

In this paper, we combine two directions of studying information propagation. We propose a feature-based model that predicts news popularity in the flow. In case we do not have social graph, we estimate the latent graph of influence and use its structural features which improve the precision and recall of prediction.

The main contribution of the paper is a method for estimating information popularity that reconstructs structural features when they are clearly not available and experimental confirmation of its efficiency on two datasets of different nature. We also showed that even if a social graph is available, it is possible to construct a much smaller graph of influence, which is an equally strong feature for predicting popularity.

The rest of the paper is organized as follows. In the second section we give brief overview of the related work. Then we present our prediction model that uses all four types of features and compensates the absence of a social graph by constructing a hidden graph of influence. Finally, experimental results are reported.

2. Related work

Area of information propagation research contains a vast amount of problem formulations. J. Cheng et al. [2] predict for the spreading information cascade whether its size will double. Y. Yang et al. [3] offer a social model, RAIN, to predict the social roles of users that influence information diffusion. J. Yang and J. Leskovec [4] model the influence of the node on the rate of diffusion. For a more detailed survey on the topic, refer to [5].

We focus on two aspects of information diffusion: propagation modelling and prediction tasks.

Propagation modelling assumes that there is hidden graph of influence between network nodes. Vertices are sources of information, while the edge (u, v) means that the source u affects v. When a social network is explicitly present we consider that information is distributed according to some algorithm in this graph. Threshold and cascade models are the most popular models of information propagation. They differ in the rules of information transfer between the nodes.

In the threshold model, edges of the graph have weights, and each user u is given a threshold value ϕ_u. User u gets the information if the sum of its active neighbours becomes equal to or greater than ϕ_u. The cascade model makes two assumptions. First, the pairwise influences of the nodes on each other are independent, and second, any user u has only one chance to transmit information to its neighbour v, regardless of the result in the next steps, the node u will not affect v. At each new iteration i, every node u that received information on the previous iteration (i-1) is trying to send it to all its neighbours v that do not have this information yet with a specified probability p_{uv}. In both models the process ends when the number of knowledgeable users does not change.

Several models were proposed for constructing an influence graph in the absence of a social graph [6-8]. Most known of them is the paper by Gomez, Leskovec and Krause [6], where authors proposed efficient algorithm for building the graph of influence called NetInf. The algorithm build a graph $\hat{G}$ that maximises probability of observed cascades:

$$\hat{G} = _{|C|\leq m}\ P(C|G),$$

where the maximisation is over all directed graphs G of at most m edges.

In case of prediction tasks, methods based on extracting features [9, 10] and deep learning methods [11-13] are used to predict how information will spread in the network based on the previous propagations. These methods do not necessarily need a social graph.

As far as we know. there is no model that combines solutions for these two directions to improve the popularity prediction. In this paper, we apply the feature-based method for solving information propagation problem and build a graph of influence if the network does not have a social graph.

3. Prediction model

In this section, we describe the proposed model. The model consists of two parts. At the first stage it reconstructs a hypothesised graph of influence using NetInf algorithm based on a given set of cascades. Then, at the prediction stage, we extract four types of features from the data and apply a classifier to predict the popularity. We use XGBoost classifier for predicting where parameters *max_depth* and *min_child_weight* were tuned using cross-validation. Other parameters were taken by default. We will use two metrics for evaluation of the model: precision and recall.

Further we consider each step in more details.

3.1 Influence graph reconstruction

Recall the assumption that some hidden relationships between the users affect the information propagation. In this work, such relations have the form of a directed graph of influence. The

nodes of the graph are information sources (such as news media or users), the edge from source u to source v means that there is an influence of u on v. In several applications, an existing social graph can be used as the influence graph. For example, in Twitter or Sina Weibo, a graph edge is defined as a subscription of one user to another. However, in most information propagation cases, an explicit social graph is absent and therefore, the influence network is hidden. In this paper, we use NetInf [6] algorithm to construct such a graph of influence.

NetInf is an iterative greedy algorithm which maximises the likelihood function to find the most cinfluential edges in the network. For a set of cascades and a graph G constructed on the involved G nodes, the probability P_C that these cascades would propagate in G, is computed. For that, the probabilities that a single cascade propagates in a subtree of G, and that the cascade propagates in G are defined. Then the likelihood function F_C equivalent to P_C is introduced, and the problem is reduced to maximising the given function at each step. At the first step an empty graph, where nodes are users of the network, is selected. Each step NetInf adds an edge with the greatest contribution to the likelihood function F_C. After m iterations we have a graph withmmost influential edges in the network. NetInf code is implemented in C++ and is publicly available [14].

3.2 Feature extraction

We extract four types of features from the data: temporal features, structural features, content features, and features of early adopters.

Temporal and structural features are collected in the same way as B. Shulman et al. [1]. Temporal features of a cascade are the most significant ones for information distribution, as was reported in the most of previous works. Temporal features are associated with the speed of propagation at the early stages. Many of such features are focused on the speed of obtaining information. The time between receiving information by the k-th early adopter and the publication in the first source is considered. Average time between information acceptance for the first half of early publications and average time for the second half of early publications are added in order to reflect propagation attenuation at the early stage.The distribution process could also be affected by the time of the first publication. At certain moments of day, news can become more popular. Therefore, we also take into account the day of the week and time of the day of the first publication.

As for the structural features, we use the reconstructed graph of influence instead of a social one.

The text of news item also have an impact on their spreading. The distribution process depends on the influence of some users on others. Usually a user writes the text similar to his influencer's. Therefore, one of the features of the news items is the similarity of texts written at the early stage. Jaccard coefficient of similarity was used.

Topic modelling is also used to study the content features. Texts of the first five publications for each information flow were used for training. Each text was preprocessed: we made all text characters lowercase, removed all stop words and non-Russian letters, then all the words were stemmed. After that, we ran 50 iterations of the LDA model [15] for 50 topics.

As a result, a vector obtained from the text of the first publication of the cascade was added to the feature vector.

3.3 Summary of selected features

Features of early adopters:

- average number of publications per day;
- early adopter id;
- percent of news written by the source in which it posted the news item at an early stage.

Temporal:

- time interval between the k-th and the first information adoption among the early adopters;
- average time between information adoption for the first half (rounded down) of early publications;
- average time between information adoption for the second half of earlypublications;
- day of the week of the early stage publications;
- time of the day of the first publication of news.

Structural:

- the number of edges of the k-th early adopter;
- the number of nodes reachable in one step from all early adopters;
- the number of edges in the subgraph of early adopters;
- the number of edges of the k-th early adopter in the subgraph of the early adopters;
- average distance between nodes in the subgraph of the early adopters.

Content:

- news topic (feature vector obtained via topic modelling);
- similarity of the text of the first and the k-th early source;
- news category.

4. Experimental evaluation

4.1 Data

The Lastfm dataset [1] was taken for the experiments. Lastfm is a music website, which have a social graph where users can listen to the music and mark the songs they like. More than 212 000 cascades were obtained for 450 000 users from the beginning of the creation of the service until February 2014. For each song a cascade is built and has the form $c = ((u_1, t_1);\ (u_2, t_2)m, \dots, (u_m, t_m))$, where u_i is the user, t_i is the time when the user liked the song.

To analyse content features and build a graph of influence, we collected 68 000 cascades for 2500 news publications at Yandex news service from January 2016. Yandex news service automatically combines news related to one topic into stories. Cascades were collected from these stories using publication date.

4.2 Experiments

Time T was taken equal 28 days for Lastfm, for Yandex $T = \infty$. The number of early adopters was taken equal to $k = 5$. For that reason, only items that have at least 5 adoptions were used in the prediction model.

NetInf is an iterative algorithm. At each iteration it adds an edge to the graph, which it considers to be the most influential. We vary the number of NetInf iterations to see how it affects the quality of the model. Results are shown on fig. 1 where X-axis corresponds to the number of NetInf iterations. One can see that the both precision and recall increase up to a certain point, then, starting from some values, it stabilises. We would recommend $m = 20\,000$ edges for this dataset as a compromise. If increase the number of edges, the model quality grows insignificantly while the overall complexity grows dramatically.

Табл. 1. Предсказание каскадов для Яндекс
Table 1. Yandex cascade prediction

Types of features	***Precision***	***Recall***
Temporal	0:685 ± 0:001	0:578 ± 0:006
Temporal + Structural	0:705 ± 0:002	0:640 ± 0:009

Temporal + Structural + Early Adopters	0:736 ± 0:002	0:675 ± 0:011
Temporal + Structural + Early Adopters + Content	**0:750** ± 0:004	**0:722** ± 0:008

We experimented with different types of features for our training model on Yandex dataset: temporal features, structural features, content ones and features of early adopters. Table 1 shows the results. At first, we used only temporal features, which were the most efficient in most of literature. When we add structural features from the graph built by NetInf, the precision and recall of the model increased by 2% and 6%, respectively. Finally, when our model is given all four types of features described in section 3.3, precision and recall improve by 7% and 15%, respectively.

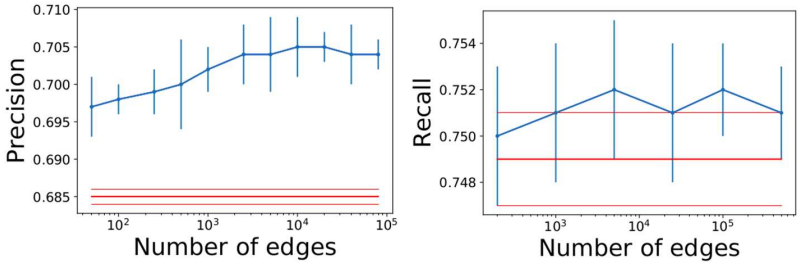

Fig. 1. Precision and recall for the predicting model using temporal and structural features extracted from the hidden graph of influence with different number of edges using Yandex cascades. Red corresponds to results of the model that use only temporal features. Blue corresponds to results of the model that use temporal and structural features.

We also checked how useful are the structural features based on the graph built by NetInf. For this purpose, we run our model on Lastfm dataset ignoring its social graph. Since the size of Lastfm social graph is more than 400 000 nodes, which was too computationally expensive for NetInf, we reduced its size. Therefore, we selected only the most active users who listened to more than 500 songs, which result to about 4000 nodes. Then we run NetInf on all the cascades from the dataset, containing only these nodes. Finally, we extracted structural features from the graph built by NetInf. We compared the effect of the obtained structural features with structural features extracted from the original social graph. Results for $m = 5000$ edges are presented in Table 2. We see that NetInf based features gives a slightly larger improvement to the prediction quality compared with social graph based features, although the difference is not significant. This means that the most influential connections between sources give the most impact on the information spread. It is also more efficient to extract structural features from the small graph of influence rather then the large social graph.

Табл. 2. Предсказание каскадов для Lastfm
Table 2. Lastfm cascade prediction

Types of features	***Precision***	***Recall***
Temporal	0:776 ± 0:003	0:749 ± 0:002
Temporal + Structural (Social graph)	0:778 ± 0:003	0:751 ± 0:002
Temporal + Structural (NetInf)	0:781 ± 0:003	0:752 ± 0:003

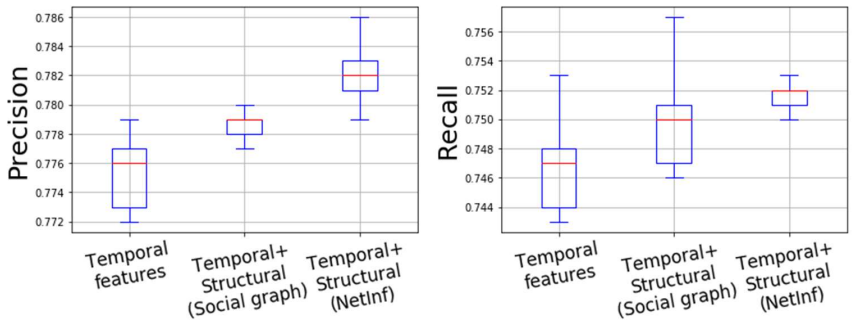

Fig. 2. Precision and recall for the predicting model using temporal, temporal+structural (social graph) and temporal+structural (NetInf) features with 100 000 edges in the hidden graph using Lastfm cascades

As with Yandex, we vary the number of NetInf iterations and observed a similar behaviour, see fig. 2. The increase of the number of iterations does not improve the prediction quality after about 5 000 edges and even starts to decrease after 100 000 edges.

5. Conclusion

We proposed a model, which predicts at the early stage whether a news story will become larger than 50% of the rest stories in the given stream of news. If the network where the news propagate, is unavailable or does not exist, the model reconstructs a graph of influence and uses it to improve the prediction quality.

We evaluated our model on the Yandex news and Lastfm datasets. Yandex news has no social graph, while the Lastfm dataset has an underlying social network. Our main results are the following.

Structural features based on a constructed graph improves the prediction precision and recall by 2% and 6%, respectively. This confirms the assumption of existence of a hidden graph of influence.

Using the NetInf algorithm allowed to achieve similar or even better prediction quality than using the original social graph. This means that instead of observing a large social graph, it is better to take some small graph containing the most influential edges that will not worsen the prediction.

Using of all four types of features (temporal, structural, early adopters, and content) significantly improves the model compared to the use of only temporal features. Precision and recall improve by 7% and 15%, respectively.

References / Список литературы

[1]. B. Shulman, A. Sharma, and D. Cosley. Predictability of popularity: gaps between prediction and understanding. In Proc. of the Tenth International AAAI Conference on Web and Social Media, 2016, pages 348–357.

[2]. J. Cheng, L. Adamic, P. A. Dow, J. M. Kleinberg, and J. Leskovec. Can cascades be predicted? In Proc. of the 23rd international conference on World Wide Web, 2014, pp. 925–936.

[3]. Y. Yang, J. Tang, C. W.-k. Leung, Y. Sun, Q. Chen, J. Li, and Q. Yang. Rain: social role-aware information diffusion. In Proc. of the Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015, pp. 367–373.

[4]. J. Yang and J. Leskovec. Modeling information diffusion in implicit networks. In Proc. of the 2010 IEEE 10th International Conference on Data Mining (ICDM, 2010), pp. 599–608.

[5]. Avetisyan A.A., Drobyshevskiy M.D., Turdakov D.Yu. Methods for Information Spread Analysis. Trudy ISP RAN/Proc. ISP RAS, vol. 30, issue 6, 2018, pp. 199-220 (in Russian). DOI: 10.15514/ISPRAS-2018-30(6)-11 / Аветисян А.А., Дробышевский М.Д., Турдаков Д.Ю. Методы анализа информационных потоков в сети Интернет. Труды ИСП РАН, том 30, вып. 6, 2018 г., стр. 199-220.

[6]. M. Gomez Rodriguez, J. Leskovec, and A. Krause. Inferring networks of diffusion and influence. In Proc. of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010, pp. 1019–1028.

[7]. M. G. Rodriguez, D. Balduzzi, and B. Schölkopf. Uncovering the temporal dynamics of diffusion networks. arXiv:1105.0697, 2011.

[8]. M. Gomez Rodriguez, J. Leskovec, and B. Schölkopf. Structure and dynamics of information pathways in online media. In Proceedings of the sixth ACM International Conference on Web Search and Data Mining, 2013, pp 23–32.

[9]. M. Jenders, G. Kasneci, and F. Naumann. Analyzing and predicting viral tweets. In Proceedings of the 22nd International Conference on World Wide Web, 2013, pp. 657–664.

[10]. S. Petrovic, M. Osborne, and V. Lavrenko. Rt to win! predicting message propagation in twitter. In Proc. of the Fifth International AAAI Conference on Weblogs and Social Media, 2011, pp. 586-589.

[11]. Q. Cao, H. Shen, K. Cen, W. Ouyang, and X. Cheng. Deephawkes: bridging the gap between prediction and understanding of information cascades. In Proc. of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 1149–1158.

[12]. C. Li, J. Ma, X. Guo, and Q. Mei. Deepcas: an end-to-end predictor of information cascades. In Proc/ of the 26th international conference on World Wide Web, 2017, pp. 577–586.

[13]. Q. Zhang, Y. Gong, J. Wu, H. Huang, and X. Huang. Retweet prediction with attention-based deep neural network. In Proc. of the 25th ACM International Conference on Information and Knowledge Management, 2016, pp. 75–84.

[14]. J. Leskovec. NETINF. Available at: http://snap.stanford.edu/netinf/, accessed 10,11.2019.

[15]. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. Journal of Machine Learning Research, No. 3, 2003, pp. 993–1022.

Информация об авторах / Information about authors

Арам Арутюнович АВЕТИСЯН, студент магистратуры факультета ВМК МГУ. Научные интересы: сбор данных, анализ информационных потоков в сети Интернет.

Aram Arutyunovich AVETISYAN, graduate student of the faculty of VMK at Moscow State University. Research interests: data collection, analysis of information flows on the Internet..

Михаил Дмитриевич ДРОБЫШЕВСКИЙ, младший научный сотрудник отдела информационных систем. Научные интересы: модели случайных графов, генерация сложных сетей с сохранением графовых свойств, машинное обучение.

Mikhail Dmitrievich DROBYSHEVSKY, Junior Researcher, Information Systems Department. Research interests: random graph models, generation of complex networks with preservation of graph properties, machine learning.

Денис Юрьевич ТУРДАКОВ, кандидат физико-математических наук, заведующий отделом информационных систем ИСП РАН, доцент кафедры системного программирования МГУ. Научные интересы: обработка естественного языка, машинное обучение, интеллектуальный анализ данных, анализ социальных сетей, распределенная обработка данных.

Denis Yuryevich TURDAKOV, Ph.D. in Physics and Mathematics, Head of the Information Systems Department at ISP RAS, Associate Professor of the System Programming Department of Moscow State University. Research interests: natural language processing, machine learning, data mining, social network analysis, distributed data processing.