Community Detection Through Topic Modeling in Social Networks

Imane Tamimi[1(✉)], El Khadir Lamrani[2], and Mohamed El Kamili[3]

[1] LIMS, FSDM, Sidi Mohammed Ben Abdellah University, Fes, Morocco
imane.tamimi1@usmba.ac.ma
[2] LTIM, FSBM, Hassan II University, Fes, Morocco
khadir.lamrani@gmail.com
[3] LIMS, FSDM, Sidi Mohammed Ben Abdellah University, Fes, Morocco
mohamed.elkamili@usmba.ac.ma

Abstract. The research on communities in social networks takes many paths in the literature, among which: the problematic of accurately detecting communities; modeling the evolution of those communities within the evolving network; and then finding the patterns that characterize this evolution over time. In our work, we focused on the problematic of detecting communities in social networks based on the information disseminated among users of the social network and the type of content shared by these users. The work at hand consists of a brief introduction to the subject and the problem definition, then we move to state the main contribution of our work which consists of a multi-layer model to detect communities of users based on the content shared by users, the lowest layer would detect topics of interest of each user while the upper layer would form communities from generated topics. We conclude the paper stating our perspectives and future works.

Keywords: Community detection · Topic modeling · Social networks

1 Introduction

With the emergence of social networks, an increasing amount of data emerges as well, making social networks one of the main providers of data and knowledge about human interactions in modern times. it is now possible for a user not only to connect and interact with another user but to share content too, thus, analyzing these content-sharing platforms is a prominent research area in social network analysis, and one fundamental theme in social network analysis focuses on the detection of communities.

Communities represent a constant source of insight and information to the scientific community and a pillar supplier of data to analysts in quite a multitude of domains, such as computer science, physics, neuroscience, telecommunications, finance, marketing, microbiology, and many others. Some of the motivations behind assessing communities within social networks are their concrete

E. Sabir et al. (Eds.): UNet 2017, LNCS 10542, pp. 70–80, 2017.
https://doi.org/10.1007/978-3-319-68179-5_7

real life applications either in sentiment analysis, or in recommender systems, or in link prediction, or in geo-localization or even fraud and terrorism detection. The list goes on.

Since the last decade, a huge amount of work has been done on detecting communities and tracking their evolution over time. The major works so far include the elaboration of algorithms that study network topologies and the structures of networks, other algorithms aim at identifying the core elements of a network and study their attributes for a better understanding of the underlying network. One main difference between algorithms then and now is that, when clustering real life networks containing millions of nodes, current algorithms are supposed to be faster with a lower complexity unlike the slower ones that are more accurate and precise.

Recent work targets both the topology of the network and the content shared to better detect communities. In fact, communities represent a great means for information diffusion. On the other hand, the type of content social network users share among them plays a key role in determining their memberships to communities. Social networks are not just graphs with nodes and edges linking the nodes, they consist of content (information) diffused, spread and shared by users.

We are interested in the issue of content shared or information disseminated in social networks and its close impact on communities formation and evolution. The issue can be defined as follows: in a social network of nodes and edges, each node interacts with the remainder of the nodes through its content or the information it diffuses. Since each node shares a different type of content, the idea is to come up with an approach that combines the type of content users have, and the information they share or adopt among themselves, from which we extract the topics to define communities in a way that is likely the most accurate.

2 Related Work

In the last decade or so, communities detection has been extensively researched and explored, there exist numerous surveys [5,10,11,23], on the subject which review various works in terms of algorithms, methods, quality measures, evaluation, benchmarks, scalability and many other aspects of communities detection over the years.

Since the problematic of community detection is a wide area in research, this section presents without going into much detail, the recent prominent work on community detection by the means of topic modeling in social networks.

The authors in [4] addressed the issue of detecting communities using topological based approaches and using topic based approaches. The authors conducted a study on the two kinds of approaches and found that the merger of the two gives the best possible outcome in terms of accuracy and semantics.

[22] introduced a type of Deep Boltzmann Machine (DBM) that is trained as a standard Restricted Boltzmann Machine (RBM), to extract distributed

semantic representations from large unstructured collections of documents. By experiments, the authors claimed their model outperformed LDA and other models on document retrieval and document classification.

[13] compared two approaches to solve the digital publishing recommender problem: latent Dirichlet allocation (LDA) and Deep Belief Nets (DBN) that represent conceptual meanings of documents and find dimensional latent representations for documents. Results have proved that Deep Belief Nets is superior in comparison to the LDA model. It manages to find better representation of documents in an output space of low dimensionality, which in turn results in fast retrieval of similar documents.

In the work of [12], the main contribution involves integrating topics into basic word embedding representation and allowing the resulting topical word embeddings to model different meanings of a word under different contexts.

[21] presented an undirected topic model used to model and automatically extract distributed semantic representations from large collections of text documents. The model is though of as different sized Restricted Boltzman Machines RBM that share parameters. The learning process of the model is easy, stable and supports documents of different lenghts. Authors demonstrated that the proposed model is able to generalize much better than LDA in terms of both the log probability on held-out documents and the retrieval accuracy.

[18] introduced a model for community extraction which incorporates both the link and content information present in the social network. The model assumes that nodes in a community communicate on topics of mutual interest, and the topics of communication, in turn, determine the communities.

[2] surveyed Topic Modeling in Text Mining under two sections. The first one presented the state of the art methods in topic modeling among which Latent semantic analysis (LSA), Probabilistic latent semantic analysis (PLSA), Latent Dirichlet Allocation (LDA), and Correlated Topic Model (CTM). The authors stated the main differences between those methods in term of characteristics, limitations and the theoretical backgrounds. The second part of the survey discussed the topic evolution models where time is taken into account. Multiple attempts to model topic evolution were listed either by discretizing time, or by using continuous time models, and some of them employ citation relationship as well as time discretization to model topic evolution.

The work of [1] presented a community detection approach which captures the content shared within the social network. The approach uses generative Restricted Boltzmann Machines model to discover communities based on the assumption that users in a community share mutual topics, the model allows users to belong to more than a community. The resulting communities were topically meaningful.

Another work [19] evaluates the impact of topic modeling in detecting communities in social networks. The authors of the paper partitioned the network into topical clusters on which a community detection algorithm was applied. The authors compared results of their method and of classic community detection. The topic oriented community detection will give better results once combined with topic analysis.

In their paper, [6] presented a model to analyze dynamic text networks. This model links network dynamics to topic dynamics through a block structure that informs both the topic assignment of a document and the linkage pattern of the network. The goal is to discover latent communities of nodes using information from both the text generated by at the nodes and the links between nodes.

[20] studied the efficiency of considering topics in detecting more meaningful communities in social networks where users express their opinions about different objects. They partitioned the network into clusters with the same topics, then they used a community detection algorithm to assess communities, compared the results with those of traditional community detection where no content has been analyzed.

[24] presented a model for community detection named RW-HDP based on Random Walk and Hierarchical Dirichlet Process topic model. They conducted at first random walks on the network and then fitted the nonparametrical topic model Hierarchical Dirichlet Process to detect community structure in order to fetch automatically the number of communities. Yet the model does not allow for the detection of overlapping communities.

3 Contributions

We propose an approach to detect communities of users by combining the information contained in the nodes and the information shared by the nodes. We develop a deep learning model to resolve the community detection problem based on topic modeling, but first we would extract shared content and then prepare and significantly represent the data (content) to fit our model; the hidden layer would detect topics of interest of each user from shared content while the upper layer would form classes of topics from generated topics using a deep learning technique. Second, we would directly extract user information and combine it with the resulted communities (defined by user memberships) so we would obtain more refined communities. Note that our model allows for a user to be part of more than a community at once. Hence the notion of overlapping communities. The following figure sums up our approach for communities detection based on content in social networks:

3.1 Data Preparation

This phase is dedicated to the preparation and analysis of the shared content structure and syntax in order represent all types of data as textual data, cleaned and grouped by nodes. our model makes usage of a series of methods used in the text mining domain.

Data Transformation. This process has a goal to represent non textual data; such as images, videos, likes, shares and so on; by the means of textual descriptions and expressions (Fig. 1).

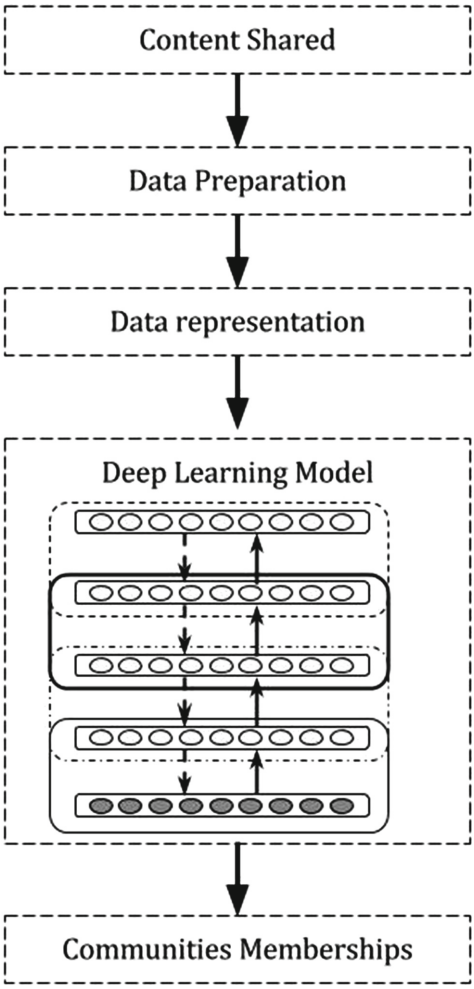

Fig. 1. Our deep learning model architecture

Preprocessing. This process has a goal to prepare the content shared by eliminating any character that does not share a useful special meaning to research information. Example: punctuation, stop words,... relating to the language used, etc.

Stemming and lemmatization. This is a preliminary operation for the recognition of words in a sentence. Indeed, It is interesting to turn all the words of the sentence in their canonical forms. We distinguish between the transformation to the root or stemme and transformation to the lemma. Lemmatisation is the name of natural language processing in the process of transforming the curls in their lemma. This process allows linguistic processing to see a lemma and its inflections semantically equivalent of fairly trivial way.

Unlike the lemma, corresponding to a real word of the language, the root or stemme is not usually a real word. The Stemming is the name given to the process that aims at transforming the flexions into their radical or stemme. Unlike lemmatisation which is therefore based on a knowledge base of inflected forms of the language to which the possible lemma is associated (called glossary), the stemmatisation only works with a basic knowledge of syntactic and grammatical rules of the language.

In the proposed system we used the lemmatisation not the stemmatisation to avoid that the words with different meaning are reduced to the same radical. For, such a transformation could distort the processing.

Group content by users: at this stage we have grouped all content shared by users, so we generate for each user, a vector that represents a set of information disseminated. each line is composed of a sequence of lemmas.

3.2 Data Representation

Most of the commonly used methods represent words in a corpus using values, thus ignore the context a word is used in. This motivates our choice for the use of Word2Vec.

Word2Vec [16,17] is a set of algorithms used for learning vector representations of words, known as word embeddings. Word2vec contains two distinct models (CBOW and skip-gram)can use either of two model architectures to produce a distributed representation of words: continuous bag of words (CBOW) or continuous skip-gram.

Word2vec has as an input a text corpus and as an output a set of vectors; feature vectors for words in that corpus. While Word2vec is not classified as a deep neural network, it turns text into a numerical form that deep networks interpret. The vectors are positioned such that those related to similar words are close in space. To achieve this, Word2vec assumes that one can determine the meaning of a word by examining its context. That said, the words appearing in the same contexts are likely similar. An extension of Word2Vec is Doc2Vec also called Paragraph2Vec, as in [15], which modifies the Word2Vec model into unsupervised learning of continuous representations for larger blocks of text, such as sentences, paragraphs or entire documents.

To implement the model, we used a training data set from "1-billion-word-language-modeling-benchmark" [3] which represents a standard training and test setup for language modeling experiments. The vectoral space of a Word2vec representation is set to a dimension equal to 300. The matrices representing the content of each user are thus of size $X_i \times 100$, X_i being the number of words shared by user i as a content.

3.3 Community Detection Using Deep Belief Nets for Topic Modeling

In this step, we present the part of the model that discovers topics using Deep Belief Nets, based on the work of [14].

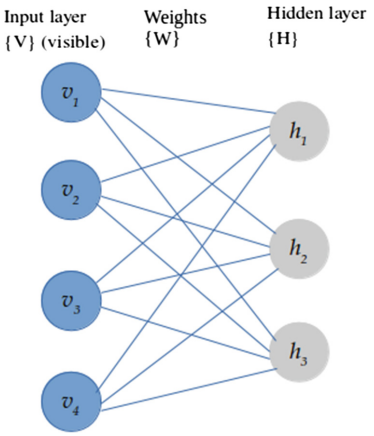

Fig. 2. Restricted Boltzman

The advantage of the DBN is that it has the ability of a highly nonlinear dimensionality reduction, due to its deep architecture. A very low-dimensional representation in output space results in a fast retrieval of similar documents to a query document. The output layer of the model groups the set of resulted topics into classes of semantically similar topics, based on the cosine distance.

A deep-belief network can be defined as a stack of Restricted Boltzman Machines [7], in which each RBM layer communicates with both the previous and subsequent layers. No lateral communication between the nodes of any single layer. To build a DBN model, Restricted Boltzmann Machine (RBM) model needs to be introduced first, which constitute the foundations of the deep learning model constructed in this article. RBMs are shallow, two-layer neural nets that represent the building blocks of deep-belief networks. The first layer of the RBM is called the visible, or input layer, the second is named the hidden layer.

Each circle in the graph below (Fig. 2) represents a node. The nodes are connected to each other through the layers, yet no node communicates nor connects with another node of the same layer. In other words, there is no intra layer communication in RBMs. Each node is a computing place that processes the input and starts by making stochastic decisions about whether or not to transmit this input. Each visible node takes a low level feature of an element of the dataset to be learned.

The visible layer reprensents the observed data and its size corresponds to the size of the data. The hidden layer represents unknown elements associated with the data and its size is randomly fixed. Depending on its size, a Restricted Boltzman Machine will be able to model more or less complex distributions.

The joint configuration (v, h) of the two layers has an energy [9] defined by:

$$E(v,h) = -\sum_{i \in visible} a_i v_i - \sum_{i \in hidden} b_j h_j - \sum_{i,j} v_i h_j w_{ij} \tag{1}$$

where v_i and h_j are the states of the visible cells i and hidden cells j, a_i, b_j are respectively their bias and w_{ij} the weights between the cells.

A probability is associated to each joint configuration through this function:

$$p(v,h) = \frac{1}{z(\theta)} e^{-E(v,h)} \tag{2}$$

z is a partition function. It represents the sum of all possible joined configurations.

$$z(\theta) = \sum_{v,h} e^{-E(v,h)} \tag{3}$$

By marginalizing on h, we obtain the probability of a visible vector v:

$$p(v) = \frac{1}{z(\theta)} \sum_{h} e^{-E(v,h)} \tag{4}$$

Based on the configuration of a layer, it is possible to know the activation probability of another cell of the second layer:

$$p(h_j = 1 \mid v) = \sigma(b_j + \sum_i v_i w_{ij}) \tag{5}$$

where σ is a logistic function (sigmoid) defined by: $1/(1 + exp(-x))$. And reciprocally:

$$p(v_i = 1 \mid h) = \sigma(a_i + \sum_j h_j w_{ij} \tag{6}$$

In [8], Hinton developed the Contrast Divergence algorithm (CD) to train the RBM network. Unlike the general sampling method, Hinton mentioned that one only needs a small sampling frequency to obtain an approximate representation when using training samples to initialize visible nodes. It quickly increases the computation speed yet keeps the precision as is.

Using the Kullback-Leibler distance to measure the difference of two probability distributions, the following formula is used to calculate:

$$CD_m = KL(p_0 \mid\mid p_\infty) - KL(p_m \mid\mid p_\infty) \tag{7}$$

By constantly updating parameters θ, CD_m converges to θ, and its precision will not change. This paper uses Contrast Divergence algorithm in RBM training, setting the value of m to 1.

The DBN consist of a visible layer, output layer and a number of hidden layers. The training process of the DBN is defined by two steps: pre-training and fine-tuning. In pretraining the layers of the DBN are separated pairwise to form restricted Boltzmann machines (RBM). Each RBM is trained independently, such that the output of the lower RBM is provided as input to the next higher level RBM and so forth. This way the layers of the DBN are trained as partly independent systems. The goal of the pretraining process is to achieve

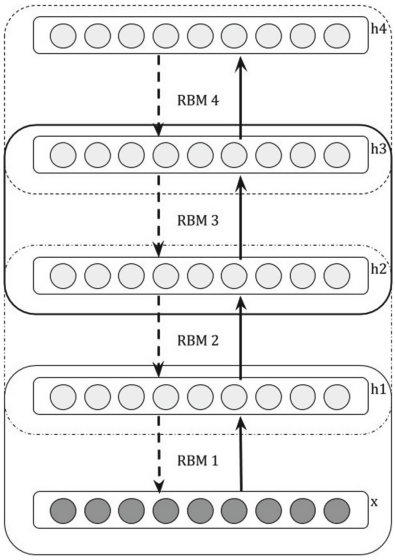

Fig. 3. The structure of DBN used in this paper

approximations of the model parameters. The model parameters from pretraining is passed on to the fine-tuning by replicating and mirroring the input and hidden layers and attaching them to the output of the DBN [14].

The structure of DBN network which is used in this paper is shown in the (Fig. 3). These networks are "limited" to a visible and a 4 hidden layers, and there are connections between the layers, but no links between the units in one layer. The hidden layer captures high data level correlation of the visible layer.

The Training Process of DBN

At the beginning of the time, by a non supervised greedy layer by layer method, the weights of the generated model are pre-trained and obtained, and Hinton has proved that unsupervised greedy layer by layer method is effective, and it is called Contrast Divergence.

The process is as follows:

1. Train the first layer as an RBM that models the raw input $x = h0$ as its visible layer. X represent vectors for users published content
2. Use that first layer to obtain a representation of the input that will be used as data for the second layer. This representation can be chosen as being the mean activations $p(h1 = 1|h0)$
3. Train the second layer as an RBM, taking the transformed data (mean activations) as training examples (for the visible layer of that RBM).
4. Iterate (2 and 3) for two layers, each time propagating upward either mean values.

The main idea of deep belief nets community detection model is to first generate a vector v among terms using word2vec distribution then passing the value to the hidden layer. In turn, the input of visual layer will be randomly selected to try to reconstruct the original input signal. Finally, these new visual neuron activation units will be transferred to reconstruct hidden layer activation unit in order to get $h1$, These back and forward steps are the familiar Gibbs sampling, and the correlation between the hidden activation unit and visual input will be the main basis for updating the weights. Next the hidden layer $h1$ is considered as an iput layer for another RBM for topic discovery phase and produce binary hidden units for topic discovery phase, and iterate for producing binary hidden units for detected community.

4 Conclusion and Perspectives

Currently, we are at the phase of elaborating the proposed model which would detect communities, based on content sharing (type of information diffused and the topics extracted from this content). We believe that the resulted communities are of high accuracy since the model allows for a user to be part of multiple communities at the same time, further future work will target the evaluation of our model with the state of the art methods of community detection and a comparison based on ground truth communities. Another possible direction of research is to study both the communities logically and topologically for more accurate results. Finally, and since this paper is a work in progress, we hope that the results we would obtain fit our theoretical hypothesis.

References

1. Abdelbary, H.A.: Semantic topics modeling approach for community detection **81**(6), 50–58 (2013)
2. Alghamdi, R., Alfalqi, K.: A survey of topic modeling in text mining. IJACSA Int. J. Adv. Comput. Sci. Appl. **6**(1), 147–153 (2015)
3. Chelba, C., Mikolov, T., Schuster, M., Ge, Q., Brants, T., Koehn, P., Robinson, T.: One billion word benchmark for measuring progress in statistical language modeling. Technical report, Google (2013)
4. Das, R., Zaheer, M., Dyer, C.: Gaussian LDA for topic models with word embeddings. Proc. ACL **2015**, 795–804 (2015)
5. Fortunato, S., Castellano, C.: Community structure in graphs. In: Computational Complexity. Theory, Techniques, and Applications 9781461418, pp. 490–512 (2012)
6. Henry, T., Banks, D., Chai, C., Owens-Oas, D.: Modeling community structure and topics in dynamic text networks. arXiv Preprint https://arxiv.org/abs/1610.05756 (2016)
7. Hinton, G.E.: A practical guide to training restricted boltzmann machines a practical guide to training restricted Boltzmann machines. Comput. (Long. Beach. Calif.) **9**(3), 1 (2010)
8. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. Neural Comput. **14**(8), 1771–1800 (2002)

9. Larochelle, H.: Classification using discriminative restricted Boltzmann machines (2008)
10. Leskovec, J., Lang, K.J., Mahoney, M.: Empirical comparison of algorithms for network community detection. In: Conference on World Wide Web WWW, pp. 631–640 (2010)
11. Liu, G.: Community structure and detection in complex networks: a survey. Cs.Gsu.Edu (2012)
12. Liu, Y., Liu, Z., Chua, T.S., Sun, M.: Topical word embeddings, pp. 2418–2424 (2015)
13. Maaloe, L., Arngren, M., Imm, O.W.I., Dk, D.T.U.: Deep belief nets for topic modeling workshop on knowledge-powered deep learning for text mining arXiv: 1501. 04325v1 [cs. CL] 18 32 (2014)., January 2015
14. Maaloe, L., Arngren, M., Winther, O.: Deep belief nets for topic modeling, 32 (2015)
15. Campr, M., Ježek, K.: Comparing semantic models for evaluating automatic document summarization. In: Král, P., Matoušek, V. (eds.) TSD 2015. LNCS, vol. 9302, pp. 252–260. Springer, Cham (2015). doi:10.1007/978-3-319-24033-6_29
16. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS, pp. 1–9 (2013)
17. Mikolov, T., Corrado, G., Chen, K., Dean, J.: Efficient estimation of word representations in vector space. In: Proceedings of International Conference on Learning Representations (ICLR 2013), pp. 1–12 (2013)
18. Pathak, N., DeLong, C., Erickson, K., Banerjee, A.: Social topic models for community extraction. In: 2nd SNA-KDD Workshop 2008 (2008)
19. Reihanian, A., Minaei-Bidgoli, B., Alizadeh, H.: Topic-oriented community detection of rating-based social networks. J. King Saud Univ. - Comput. Inf. Sci. pp. 1–8 (2015)
20. Reihanian, A., Minaei-Bidgoli, B., Alizadeh, H.: Topic-oriented community detection of rating-based social networks. J. King Saud Univ. - Comput. Inf. Sci. **28**(3), 303–310 (2016)
21. Salakhutdinov, R., Hinton, G.: Replicated softmax: an undirected topic model, pp. 1–8
22. Srivastava, N., Hinton, G.: Modeling documents with a deep Boltzmann machine
23. Xie, J., Kelley, S., Szymanski, B.K.: Overlapping community detection in networks: the state-of-the-art and comparative study. ACM Comput. Surv. **45**(4), 43:1–43:35 (2013)
24. Zhu, R., Jiang, W.: Combing random walks and nonparametric bayesian topic model for community detection, pp. 1–13 (2016)