Semantic-based Tag Recommendation in Scientific Bookmarking Systems

Hebatallah A. Mohamed Hassan
Roma Tre University
Rome, Italy
hebaatef.ibrahim@gmail.com

Giuseppe Sansonetti
Roma Tre University
Rome, Italy
gsansone@dia.uniroma3.it

Fabio Gasparetti
Roma Tre University
Rome, Italy
gaspare@dia.uniroma3.it

Alessandro Micarelli
Roma Tre University
Rome, Italy
micarel@dia.uniroma3.it

ABSTRACT

Recently, tagging has become a common way for users to organize and share digital content, and tag recommendation (TR) has become a very important research topic. Most of the recommendation approaches which are based on text embedding have utilized bag-of-words technique. On the other hand, proposed deep learning methods for capturing semantic meanings in the text, have been proved to be effective in various natural language processing (NLP) applications. In this paper, we present a content-based TR method that adopts deep recurrent neural networks to encode titles and abstracts of scientific articles into semantic vectors for enhancing the recommendation task, specifically bidirectional gated recurrent units (bi-GRUs) with attention mechanism. The experimental evaluation is performed on a dataset from CiteULike. The overall findings show that the proposed model is effective in representing scientific articles for tag recommendation.

KEYWORDS

Tag Recommendation; Scientific Bookmarking Systems; Deep Learning; Attention Mechanism

ACM Reference Format:
Hebatallah A. Mohamed Hassan, Giuseppe Sansonetti, Fabio Gasparetti, and Alessandro Micarelli. 2018. Semantic-based Tag Recommendation in Scientific Bookmarking Systems. In *Twelfth ACM Conference on Recommender Systems (RecSys '18), October 2–7, 2018, Vancouver, BC, Canada.* ACM, New York, NY, USA, Article 4, 5 pages. https://doi.org/10.1145/3240323.3240409

1 INTRODUCTION

Recently, tagging systems have become popular mean for users to contribute to the annotation of large corpora. By means of tags, users can conceptually organize and summarize information, but also search for it. TR in such systems help users find appropriate tags for resources and help consolidate annotations across all users and resources.

Existing TR methods can be categorized into three main classes: content-based methods, collaborative filtering methods, and hybrid methods [2]. In this paper, we focus on content-based methods, where object's textual features, such as title, description, and user comments are used for training the TR systems through multi-label learning algorithms. The most popular learning methods for content-based method is to represent each document through a vector of all word occurrences weighted by term frequency-inverse document frequency (TF-IDF), also statistical topic modeling techniques, such as Latent Dirichlet Allocation (LDA) [5, 9, 10], is commonly used. However, those methods do not utilize information such as text order and semantic of words. On the other hand, deep learning has shown great potential for learning effective representations and delivered state-of-the-art performance on various NLP tasks. Liu *et al.* [13] have utilized convolutional neural networks (CNNs) [8] for extremely large label collection. In [3], the author has explored how both a CNN and a GRU can independently be used with pre-trained word embeddings to solve a large scale multi-label text classification problem, where the designed models have been tested on scientific abstracts from PubMed [1].

On the other hand, attention-based models have demonstrated success in a wide range of NLP tasks. It was originally proposed in machine translation tasks to deal with the issue for encoder-decoder approaches, that all the necessary information should be compressed into fixed length encoding vector [1]. Then, an attention model is leveraged for generating image descriptions [22]. Other attention-based work includes sentence summarization [18]. Yang *et al.* [23] exploit attention in neural networks, enabling it to attend differentially to more and less important content when constructing the document representation, by capturing hierarchical patterns of documents from word to sentence and finally to the whole document.

In this work, we propose an approach that adopts bidirectional gated recurrent units (bi-GRUs) with attention mechanism to capture important patterns and semantic representations of summaries of scientific articles (i.e., titles and abstracts) for the TR task. We performed experiments to investigate the influence of attention layers in our model. Additionally, we compared the proposed method

[1] www.ncbi.nlm.nih.gov/pubmed

RecSys '18, October 2–7, 2018, Vancouver, BC, Canada

ACM ISBN 978-1-4503-5901-6/18/10…$15.00
https://doi.org/10.1145/3240323.3240409

with several baselines. Preliminary experimental results on a real dataset extracted from CiteULike [2] prove the validity of the proposed approach. To the best of our knowledge, this is the first study utilizing attention mechanism to represent research papers (i.e., their titles and abstracts) for a tag recommendation task.

2 METHODOLOGY

2.1 Problem Definition

Given a set of N documents $\{d_i\}$ and a vocabulary of M tags $\{t_j\}$, we consider a dataset $D = \{(x_i, y_j)\}$, where x_i is the keyword-based representation of the document d_i, and y_j assumes 1 or 0 value if the tag t_j is associated with d_i or not, respectively. The representation x_i consists of a sequence of d-dimensional embeddings of consecutive words grouped into sentences, as follows:

$$x_i = \{\{w_{1,1}, ..., w_{1,N_T}\}, \cdots, \{w_{N_K,1}, ..., w_{N_K,N_T}\}\} \quad (1)$$

being N_T the maximum number of words in a sentence, and N_K the maximum number of sentences in a document. The network takes as input a document x_i and outputs a document vector u_i. The output u_i is used by the classification layer to determine $\{y_j\}$, that is, the tags related to d_i.

2.2 The Proposed Approach

To solve the aforementioned problem, we adopt a general hierarchical attention architecture for document representation, as shown in Figure 1. The following sections show the details of different components.

2.2.1 Word Embeddings. Titles and abstracts of scholarly documents are extracted, concatenated, and subjected to tokenization and stop word removal. Each word in the document is represented as a fixed-size vector from pre-trained word embeddings. We used GloVe [16] word embeddings for this purpose.

2.2.2 Bidirectional GRU Encoders. Gated recurrent units (GRUs) use reset and update gate vectors at each position to control the information flow along the sequence, thus improving the modeling of long-range dependencies. Bidirectional GRU [19] is another version of GRU. Unlike standard GRUs, which only capture information from the current and past states, bidirectional GRU determine outputs also considering inputs from the future. At word level, we embed each word in a sentence into a low dimensional semantic space using a bi-GRUs. At sentence level, we also feed the sentence embeddings into bi-GRUs and then obtain the document representation. At word level, the function g_w encodes the sequence of input words $\{w_{l,t}|t = 1, ..., N_T\}$ for each sentence l of the document, that is:

$$h_w{}^{(l,t)} = \{g_w(w_{l,t})|t = 1, ..., N_T\} \quad (2)$$

At sentence level, after combining the intermediate word vectors $\{h_w{}^{(l,t)}|t = 1, ..., N_T\}$ to a sentence vector s_l, the g_s function encodes the sequence of sentence vectors $\{s_l|l = 1, ..., N_K\}$, that is, $h_s{}^{(l)}$.

The g_w and g_s functions are bidirectional GRUs with parameters H_w and H_s respectively, obtained from the forward GRU, $\vec{g_w}$, and the backward GRU, $\overleftarrow{g_w}$:

$$h_w{}^{(l,t)} = [\vec{g_w}(h_w^{(l,t)}); \overleftarrow{g_w}(h_w^{(l,t)})] \quad (3)$$

The same concatenation procedure is applied to the hidden state representation of a sentence $h_s{}^{(l)}$.

2.2.3 Attention Layers. A typical way of assigning a representation to a given word sequence at each level is by taking the last hidden-state vector output by the encoder. However, it is hard to encode all the relevant input information needed in a fixed-length vector, which may limit the performance of these networks. In addition, not all the input words contribute equally to the representation of the sentence meaning, and not all the sentences contribute equally to the document representation. This problem is addressed by an attention mechanism at each level, denoted by α_w and α_s, that estimates the importance of each hidden state vector with respect to the sentence or document meaning, respectively. The sentence vector $s_l \in R^{dw}$, where dw is the dimension of the word encoder, is thus obtained as follows:

$$\sum_t \alpha_w^{(l,t)} h_w^{(l,t)} = \frac{exp(u_{l,t}^\top u_w)}{\sum_t exp(u_{l,t}^\top u_w)} h_w^{(l,t)} \quad (4)$$

where $u_{l,t} = f_w(h_w^{(l,t)})$ is a fully-connected neural network with W_w parameters. Similarly, the document vector $u \in R^{ds}$, where ds is the dimension of the sentence encoder, is obtained as follows:

$$\sum_l \alpha_s^{(l)} h_s^{(l)} = \frac{exp(u_l^\top u_s)}{\sum_l exp(u_l^\top u_s)} h_s^{(l)} \quad (5)$$

We feed the output vector to a linear layer whose output length is M, the cardinality of the tag vocabulary.

2.2.4 Classification Layer. Finally, we formulate the tag recommendation task as a multi-label classification problem. We train our model in a supervised manner by minimizing the cross-entropy error of the tag classification. We adopted a softmax classifiers of predefined classes on top for classification, and the input is a combination of the features generated from the document level attention.

3 EXPERIMENTS

3.1 Dataset

CiteULike is an online platform which allows users to create personal libraries by saving papers that are of interest to them. We used citeulike-a dataset that is collected from [21]. It consists of 169804 papers with 46391 tags. We selected the 10 most common tags from the dataset and extracted the papers (titles & abstracts) that have at least one of those tags. The statistics of our dataset are shown in Table 1.

Table 1: Statistics of the evaluation dataset (tag cardinality is the average number of tags assigned to a document).

#Documents	#Tags	#Vocabulary	Tag Cardinality
6397	10	8002	4

[2] http://www.citeulike.org

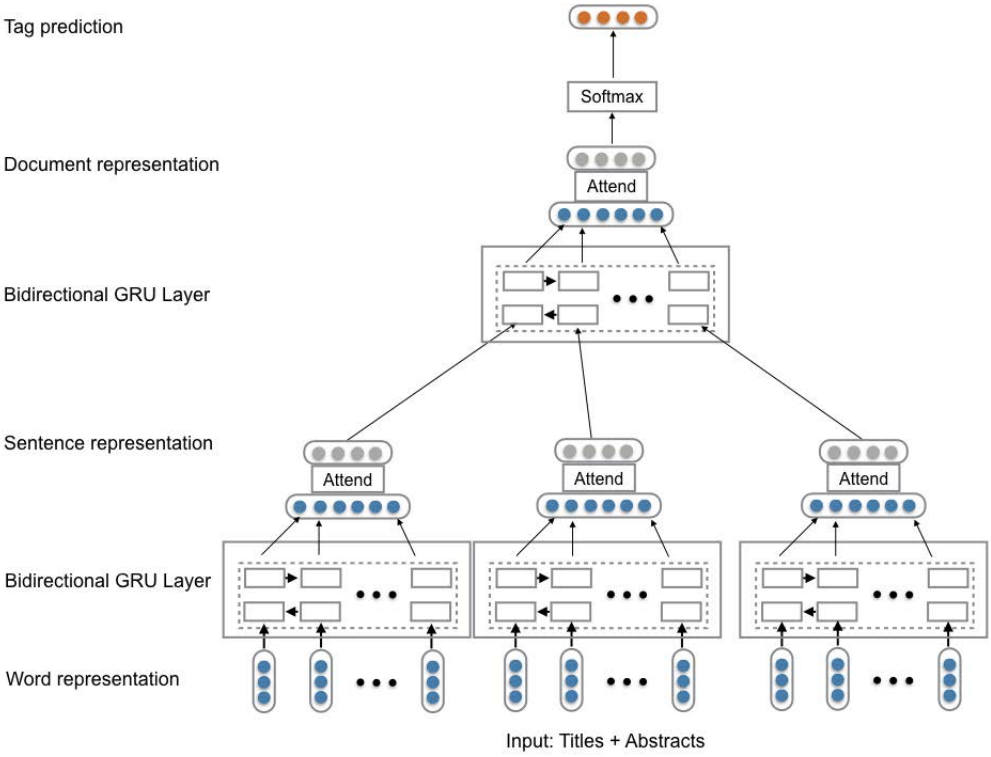

Figure 1: The Proposed Approach.

3.2 Evaluation

To evaluate the performance, we considered *Micro-recall*, *Micro-precision*, and *Micro-F1* measures. Micro-averaging is a commonly used method in Information Retrieval. It is essential in any classification task such as social tag prediction, that involves imbalanced classes.

3.3 Comparisons

In order to perform an empirical evaluation of the proposed method (i.e., **Bi-GRU+Att**), we made a comparison with the following baseline methods in text-based multi-label classification:

- **Multinomial Naïve Bayes (NB):** since hashtag recommendation task is formalized as a classification task, we applied NB [14] to model the probability of each tag given the textual information of the papers (titles and abstracts).
- **Support Vector Machine (SVM):** a widely used supervised text classifier [6]. We used one-vs-rest scheme with bag-of-words (BOW) features.
- **Latent Dirichlet Allocation (LDA):** it has been utilized widely in tag recommendation [7]. Similar to [20], we trained a n-topic LDA model [4], where n is the number of tags.
- **Paragraph-Vector:** a state-of-the-art performer on several benchmark datasets for semantic representation of paragraphs [11] .

3.4 Experimental Settings

We implemented our model using the Python open source library Keras [3] with a TensorFlow backend. We run the experiments on a GPU server, NVIDIA Tesla 100 GPU. We repeat the evaluation 5 times, with randomly separating our data into 90% and 10% for training and testing, and the average performance is reported. Titles and abstracts of each research paper were concatenated, tokenized, and subjected to stop word removal before the lemmatization performed by the NLTK package [4]. The maximum number of sentences per document was set to 10, and the maximum number of words per sentence to 50 with zero-pad the beginning of sentences and documents, if necessary. Furthermore, we used pre-trained GloVe-6B-300D vectors [16] for word-level embeddings.

We added an additional dense layer after the document attention layer, in order to increase the complexity of the network. We also added drop out layers before and after the dense layer in order to prevent overfitting. Our method achieved the best performance when the dimension of hidden state of bi-GRU networks was set to 25 and 5 for sentence and paragraph encoding respectively. Additional hyperparameters were 50 hidden states for the dense layer and 0.2 as dropout value. A mini-batch stochastic gradient descent (SGD) algorithm was used to train our model [12]. Batch size was

[3] https://github.com/fchollet/keras
[4] http://www.nltk.org

set to 64 to minimize the loss function of a categorical cross entropy. Moreover, we manually set the prediction threshold to 0.5. Finally, the epoch was set to depend on an early stop, which relied on a validation set to decide when to stop the training. In our experiments, it took 5 epochs to stop the training process.

For NB, multinomial NB implementation in the popular scikit-learn library [15] has been used, with bag-of-unigrams for representing text features. In SVM, we used TF-IDF for representing text features and trained a set of SVM linear classifiers, one for each label, using the OneVsRestClassifier available in the scikit-learn library. For LDA, we used the gensim [17] LDA implementation in our experiments using TF-IDF representation, and we tested with different numbers of LDA topic size K, thus finding that $K = 10$ was an optimal setting. And the best performing Paragraph Vector model was with 200 dimensions, and a context window size of 3.

3.5 Results

We validated the performance based on the exploited dataset, and used the same model weights on the test dataset to obtain the experimental results shown in Table 2.

Table 2: Comparison analysis on top-10 tag prediction task.

Method	Micro-Recall	Micro-Precision	Micro-F1
NB	0.03	0.21	0.05
SVM	0.05	0.20	0.09
LDA	0.16	0.20	0.17
Paragraph Vector	0.24	**0.22**	0.22
Bi-GRU+Att	**0.44**	0.20	**0.28**

The following observations can be drawn by analyzing the obtained outcomes: (i) LDA and Paragraph Vector performed better than NB and SVM, showing that they could capture more semantic information; (ii) the attention layers improves the Micro-F1 score, which shows the effectiveness of focusing on important words and sentences for this task.

4 CONCLUSION AND FUTURE WORK

In this paper, we have proposed an attention-based bidirectional gated recurrent unit (bi-GRU) model for the tag recommendation task in scholarly materials. We formulated our problem as a multi-label classification task, and utilized hierarchical word and sentence level attention networks for aggregating important words and sentences, in order to increase the general representation and visualization of the key concepts in research papers. The results of our experiments on a CiteULike dataset show that the proposed approach outperforms some state-of-the-art methods. In the future, more extensive experiments will be conducted to further evaluate the performance of our proposed approach on different parameter settings and larger datasets. We also aim to extend our work to explore multi-label ranking methods. Finally, using a model to learn an optimal threshold for rounding the probabilities for tag recommendation may also lead to increased performance.

REFERENCES

[1] D. Bahdanau, K. Cho, and Y. Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *ArXiv e-prints* (Sept. 2014). arXiv:cs.CL/1409.0473

[2] Fabiano M Belém, Jussara M Almeida, and Marcos A Gonçalves. 2017. A survey on tag recommendation methods. *Journal of the Association for Information Science and Technology* 68, 4 (2017), 830–844.

[3] Mark J. Berger. 2015. *Large Scale Multi-Label Text Classification with Semantic Word Vectors.* Technical Report. Stanford University.

[4] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* 3 (March 2003), 993–1022. http://dl.acm.org/citation.cfm?id=944919.944937

[5] Beomseok Hong, Yanggon Kim, and Sang Ho Lee. 2017. An Efficient Tag Recommendation Method Using Topic Modeling Approaches. In *Proceedings of the International Conference on Research in Adaptive and Convergent Systems (RACS '17)*. ACM, New York, NY, USA, 56–61. https://doi.org/10.1145/3129676.3129709

[6] Jens Illig, Andreas Hotho, Robert Jï£¡schke, and Gerd Stumme. 2011. A Comparison of Content-Based Tag Recommendations in Folksonomy Systems. In *Knowledge Processing and Data Analysis (Lecture Notes in Computer Science)*, Karl Erich Wolff, Dmitry E. Palchunov, Nikolay G. Zagoruiko, and Urs Andelfinger (Eds.), Vol. 6581. Springer, Berlin/Heidelberg, 136–149. https://doi.org/10.1007/978-3-642-22140-8_9

[7] H. Jelodar, Y. Wang, C. Yuan, and X. Feng. 2017. Latent Dirichlet Allocation (LDA) and Topic modeling: models, applications, a survey. *ArXiv e-prints* (Nov. 2017). arXiv:cs.IR/1711.04305

[8] Y. Kim. 2014. Convolutional Neural Networks for Sentence Classification. *ArXiv e-prints* (Aug. 2014). arXiv:cs.CL/1408.5882

[9] Ralf Krestel and Peter Fankhauser. 2009. Tag Recommendation Using Probabilistic Topic Models. In *Proceedings of the 2009th International Conference on ECML PKDD Discovery Challenge - Volume 497 (ECMLPKDDDC'09)*. CEUR-WS.org, Aachen, Germany, Germany, 131–141. http://dl.acm.org/citation.cfm?id=3056147.3056158

[10] Ralf Krestel, Peter Fankhauser, and Wolfgang Nejdl. 2009. Latent dirichlet allocation for tag recommendation. In *Proceedings of the third ACM Conference on Recommender Systems (RecSys '09)*. ACM, New York, NY, USA, 61–68. https://doi.org/10.1145/1639714.1639726

[11] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning*. 1188–1196.

[12] Mu Li, Tong Zhang, Yuqiang Chen, and Alexander J. Smola. 2014. Efficient Mini-batch Training for Stochastic Optimization. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)*. ACM, New York, NY, USA, 661–670. https://doi.org/10.1145/2623330.2623612

[13] Jingzhou Liu, Wei-Cheng Chang, Yuexin Wu, and Yiming Yang. 2017. Deep Learning for Extreme Multi-label Text Classification. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. ACM, New York, NY, USA, 115–124. https://doi.org/10.1145/3077136.3080834

[14] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval.* Cambridge University Press, New York, NY, USA.

[15] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* 12 (Nov. 2011), 2825–2830. http://dl.acm.org/citation.cfm?id=1953048.2078195

[16] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation.. In *EMNLP*, Vol. 14. 1532–1543. https://nlp.stanford.edu/pubs/glove.pdf

[17] Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. ELRA, Valletta, Malta, 45–50.

[18] Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685* (2015).

[19] M. Schuster and K.K. Paliwal. 1997. Bidirectional Recurrent Neural Networks. *Trans. Sig. Proc.* 45, 11 (Nov. 1997), 2673–2681. https://doi.org/10.1109/78.650093

[20] Yang Song, Lu Zhang, and C. Lee Giles. 2011. Automatic Tag Recommendation Algorithms for Social Recommender Systems. *ACM Trans. Web* 5, 1, Article 4 (Feb. 2011), 31 pages. https://doi.org/10.1145/1921591.1921595

[21] Hao Wang, Binyi Chen, and Wu-Jun Li. 2013. Collaborative Topic Regression with Social Regularization for Tag Recommendation.. In *IJCAI*. 2719–2725.

[22] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention.. In *ICML (JMLR Workshop and Conference Proceedings)*, Francis R. Bach and David M. Blei (Eds.), Vol. 37. JMLR.org, 2048–2057. http://dblp.uni-trier.de/db/conf/icml/icml2015.html#XuBKCCSZB15

[23] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J. Smola, and Eduard H. Hovy. 2016. Hierarchical Attention Networks for Document Classification. In *HLT-NAACL*.