2015 IEEE International Conference on Smart City/SocialCom/SustainCom together with DataCom 2015 and SC2 2015

Extracting Image Context from Pinterest for Image Recommendation

Philipp Berger*, Patrick Hennig*,
Daniel Dummer†, Alexander Ernst†, Thomas Hille†, Frederik Schulze† and Christoph Meinel‡
Hasso-Plattner-Institute
University of Potsdam, Germany
*{philipp.berger,patrick.hennig}@hpi.de
†{daniel.dummer, alexander.ernst,thomas.hille, frederik.schulze}@student.hpi.de
‡office-meinel@hpi.de

***Abstract*—Image search and recommendation engines try to extract relevant images for a user's information need. Existing approaches use manual tags of networks like Flickr or the surrounding webpages to create context to foster the search. Pinterest as a new upcoming social bookmarking service allows us to gain more context for an image than before. By using board headline, pin descriptions, and the actual content of the bookmarked pages we build a much more complex context.**

As a use case, we recommend images for blog articles to show the feasibility of the context of Pinterest. We apply tag-based retrieval models to actual propose matching images for article texts. This enables blog authors to get image suggestions for their articles to speed up the creation of appealing articles. Our evaluation shows that a retrieval model based on cosine similarity yields promising results. Given the bookmarked pages, it reaches a precision of 96% to predict the pinned images. Further, a user survey yields that the recommended images are actual usable for the articles.

I. Introduction

Sharing images on the internet and, especially, via social media is gaining more and more growth. For example, Facebook receives 300 million image uploads per day [1](Aug.2012). As the amount of images keeps growing, web image search interfaces (e.g. Google Image Search[1]) gain more importance. Thoses interface mainly support two tasks: search by keyword terms and search by image. Given an image, search engine return similar image in terms of visual properties like textures and color. To search for a keyword, the search engines also index the surrounding text of images [2] and user-defined tags (e.g. extracted from Flickr [3]) . Further, browsers for massive image collection enable users to cluster images according to their visual similarity or shared semantic concepts [4].

Especially professionals (e.g. journalists, bloggers, directors) face the challenge of retrieving matching images to their articles. Hereby, the authors could use the immense collections of images that exists in social networks. Nevertheless, it is difficult for the authors to formulate keywords for search engines to retrieve images. Those, they rely on browsing image collections supported by intelligent clustering.

We provide authors a tool that dynamically propose images based on the already writing text of an author. By converting the user's text into a set of matching keywords we are able to input this into a keyword based search engine. As the textual context of current engines is limited to the surrounding text of images or user-defined tags, we explore the usability of *Pinterest*[2] to broaden the index context of images.

Pinterest is a fast growing social network specialized on webpage bookmarking. Each user has the ability to bookmark pages via posting an descriptive image. Pinterest had more than 5 million images uploaded per day in 2013 [5]. Although this is just a fraction of the uploaded images to Facebook the perceived quality is better. This is caused by the collective nature of Pinterest. Each users tries to find the best descriptive image for a webpage to receive more shares (called repins). Pinterest's users can share image via so called *pins*, which can be seen as bookmarks to webpages. Each *pin*(see Fig. 1) so consists of:

- the link to the external webpage
- an image which can mostly be found on the webpage the pin points to
- a pin description, filled by the user (often generated from the page title)
- multiple *boards* containing the pin
- a number of *repins*

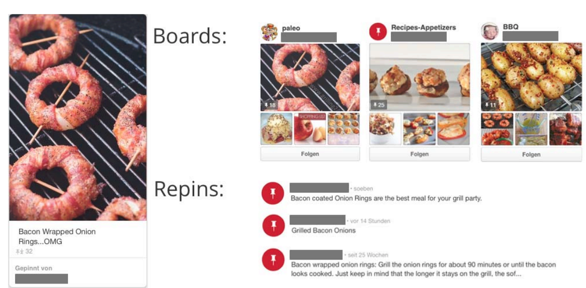

Fig. 1. Sample for pin information from board names and repins.

In this paper, we evaluate the diverse textual elements given by Pinterest to build the required textual context for indexing. We describe how this information can be used to extract matching

[1]https://images.google.com/

[2]https://www.pinterest.com/

978-1-5090-1893-2/15 $31.00 © 2015 IEEE
DOI 10.1109/SmartCity.2015.92

keywords. Similarly, we show how to extract keywords from the user's article text to finally evaluate which retrieval model like BM25 or cosine similarity can be used to find images.

II. Related Work

General analysis of social networks and image recommendation is the subject of several scientific papers. Nevertheless, Pinterest as a new type of social network is only sparsely investigated.

The majority of image retrieval research performs image analysis to retrieve visual properties [6], [7] or concepts [8]. Further keyword-based image retrieval focuses on the usage of text surround the image on a webpage. Shen et al. [9], [10] described the use of Weight ChainNet which are based on lexical chain that represents the semantics of an image from its surrounding text. The results are with 0.5 Precision at 0.5 recall fairly good since the user can directly detect relevance of an image.

Keyword-based search in general is researched in the area of social network search. Zhao et al. [11] show how to extract content of the social network Twitter. They proposed to use a context-sensitive PageRank method for keyword ranking. Marlow et al. [12] analyzed the tagging on the online photo service Flickr and recommend tags to assist the user. Instead of content analysis they propose a co-occurrence analysis of tags. We use these concepts to extract keywords from the textual context of Pinterest and from the authors text. These keywords are the basis of the used retrieval models.

BM25 [13] is a ranking function used for retrieval of documents for search queries. The bulk of work related to BM25 focuses on the adaption and improvement of the ranking for specialized search tasks for documents [14]–[17]. Equally, the cosine similarity is used in information retrieval [18]. It is essentially a vector distance used to compare document and query vectors to rank search results accordingly.

Nevertheless, studies focusing on Pinterest mainly focus on the understanding of this social network instead of using it. Mittal et al. [19] characterized Pinterest in multiple aspects like prominent topics, top image sources and geographical distribution. These basic information about the structure of Pinterest was helpful as starting point for this research. Ottoni et al. [20] investigates the gender roles in the network. They discovered that Pinterest is mostly used by women and therefore also discovered the most used topics.

III. Concept

We introduce a image recommendation engine that uses the textual context of Pinterest to recommend images for article texts. Figure 2 visualizes the procedure used in this work. After crawling Pinterest to build a sufficient data set we retrieve and analyze the textual context of the pinned images. This is done by combining the text extracted from the pin title, description, board titles, and the text content of the pinned webpage. For the sake of simplicity, we focus on English text element because most of the text processing steps are best fitted for English. After filter the image context, we apply part-of-speech tagging and word stemming (lemmatization) to retrieve the tags. To estimate the relative importance of tags per image context, we apply two approaches: *tf/idf* scoring [21] and *TextRank* [22]. Our search index contains the score tag vectors for each image of our crawled Pinterest dataset. During the writing of an article the full text gets continuously transfer to the recommendation system. The text undergoes the same tag extraction procedure as the textual image. For recommending the images, we use the BM25 and cosine similarity models that outputs a ranked lists of images.

A. Image Source Pinterest

As an image-focused social network, Pinterest is a good source for information about the image and especially about its context [23]. Additional to the pure image, Pinterest provides numerous user maintained chunks of information for these images: the description of the pin, the board name, an external url and could be extended by several repins that add further descriptions of that image. For example the same image of *Bacon Coated Onion Rings* could be described as *Grilled Bacon Onions* or *Bacon Wrapped Onion Rings* as displayed by figure 1. Additionally most repins are included in different boards with various names, e.g. "Food", "Grilled" or "Bacon Recipes". Furthermore the pin is linked to an external website that contains the whole receipt.

To combine the various textual element into one image context, we need to identify a proper weighting scheme. Therefore we considered the denseness, diversity and uniqueness of the text. For example, it is likely that similar boards have a similar or even the same name, which is often inspired by the category. This textual information is very dense, but not unique for each pin because a boards contains multiple pins. Although copied during the process of re-pinning, the description of a pin is rather unique. The information extracted from the bookmarked webpage is very unique, but not as dense as other elements.

Given the vast amount of images provided by Pinterest, we focus on the precision of our image retrieval. Thus, dense and unique information are by fare more relevant to us. As a result the pin description is the most valuable chunk of information we can extract from Pinterest (ignoring duplicates). Due to the low density of information, the external website is not as important as the pin description, but also allows us to get a wider description of an image.

After the collecting and weighting of information it is essentiell to find a suitable representation. The representation as tags has the primary advantage that it is possible to reuse existing retrieval models for querying. It is necessary to filter for most descriptive words to generate tags. Since most images in Pinterest show an object that the user is interested in [24], nouns and adjectives are more relevant than verbs. To get the most descriptive words, the text is analyzed with Textrank or tfidf and afterwards filtered for nouns and adjectives. These tags are weighted by their source and stored in a database with references to the image source url.

B. Querying Pinterest

Pinterest is a bookmarking site that refers to external sites by showing a representative image of the content. As a result, the image of an external site is the most important factor for baiting Pinterest users to visit that external site. Thus, Pinterest is a promising source of images for recommending image for

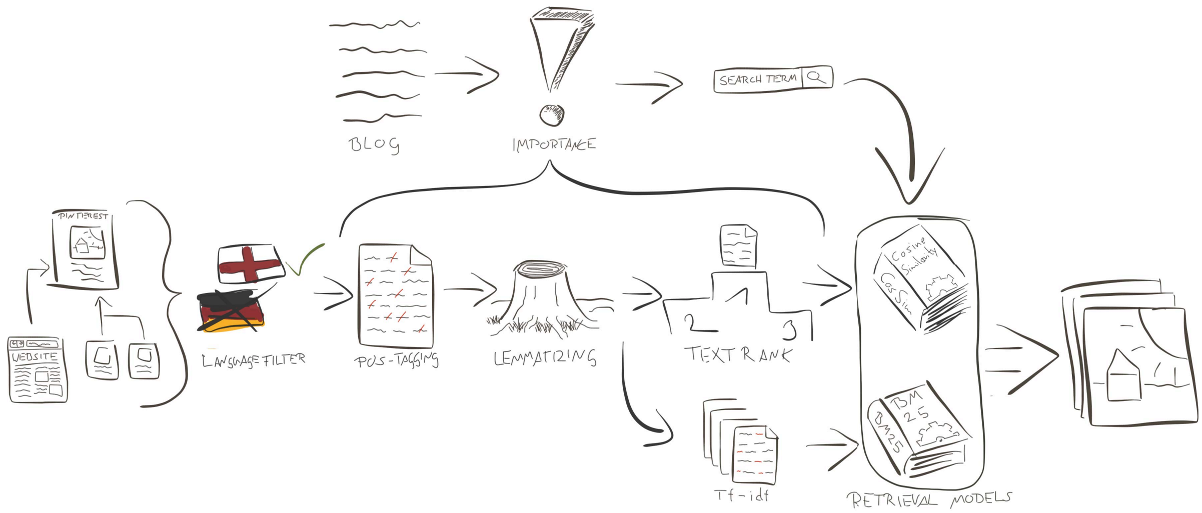

Fig. 2. After extracting the textual context of Pinterest, we apply the visualized tag extraction procedure. A similar procedure is also used to create query tags out of the input text. Finally, we compare both tag sets using the retrieval models to propose relevant images.

full article like blog posts. In order to do so, we also analyze the query text with Textrank to get the important nouns and adjectives of that text. Although the text of a website is much longer than a pin description, we extract an equal number of tags by considering only the most important terms using TextRank and tfidf. With that information and an established retrieval model like cosine similarity the system can find a relevant image for the text.

IV. Implementation

In this section the implementation of the whole system is explained. An overview of the complete system can be found in figure 2. The system can be split into tree steps.
At first an image repository is built up by extracting pins from Pinterest and generating tags for the images of the pins. As the result the system has a big amount of images with annotated tags, which have different scores depending on their importance. This will be explained in the sections IV-A and IV-C. The first step has to be done in advance before the image retrieval system can be used. It can be triggered anytime to extend the database.
The top part of the flow in figure 2 shows the second step. To retrieve images for full-text, the system extracts important keywords - similar to the extraction of tags for the images, which is further explained in section IV-D.
In the third step the extracted keywords are taken as an input for the image retrieval system described in IV-E.

A. Getting the Data

Since Pinterest has no official API to access their data until today, the pins and boards were extracted using a web-crawler. We use the Apache Nutch[3] project for our crawler, because it is highly scalable (using Hadoop[4]) and easily adaptable using custom extracting and storage components. To retrieve the Pinterest-specific element like board names, descriptions, pin titles etc. we implemented a nutch plugin. Pinterest's HTML source is highly enriched with semantic meta tags. We use severall tags to create our data set (see Table I).

meta property	description
og:type	give the type of a page (eg. board, pin or user)
pinterestapp:pinner	gives the link to the author page
og:description	contains the actual pin/board description
pinterestapp:repins	number of repins of a pin
pinterestapp:likes	number of like of a pin
pinterestapp:pinboard	the board a pin is contained in
pinterestapp:source	link to the bookmarked webpage
og:image	actual link to the bookmarked image
pinterestapp:pins	number of pins of a user
pinterestapp:followers	number of follower of board/users
pinterestapp:category	categories of board or pin
pinterestapp:boards	number of user's boards
pinterestapp:following	number of user's followed boards
pinterestapp:about	user description
pinterestapp:facebook	user's facebook profile link
pinterestapp:twitter	user's twitter feed link

TABLE I. HTML META PROPERTIES USED FOR EXTRACTION

Beside the meta tags, we also preserved the network structure of Pinterest. Hereby, we have to use regular expression to extract the corresponding http links from the Pinterest pages. Those links are not semantically annotated. An example regular expression for collecting repins is:

```
^.*/pin/[0-9]*/repins/?\\s*$.
```

After crawling we store the data into an relational database for later retrieval and analysis. One major challenge during our data collection is the call limitation of Pinterest. Therefore, the current work is limited to a test set of 100 000 pins.

B. Textrank

The TextRank [22] is a Graph-based ranking algorithm that can be applied to a variety of natural language applications. In this system it is used for the extraction of keywords, as it provides better results than pure term frequency.
The TextRank algorithm builds a graph with words as vertices

[3] https://nutch.apache.org/
[4] https://hadoop.apache.org/

and connects them with edges if they are written next to each other within a certain distance. To calculate the TextRank, it starts with a base value for every vertex. By iterating over each node it calculates its new score based on its own score and the scores of the connected vertices, by using the formula displayed in equation 1. This calculation is done round-wise until the TextRank is nearly constant for each vertex.

$$S(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{1}{|Out(V_j)|} S(V_j) \tag{1}$$

Since the system needs to weight different parts of text higher then others, the algorithm needs to be extended with weighting of vertices. For example is a headline more dense and should therefore be higher weighted than the rest of the text. To realize this, the formula for calculating the TextRank is adapted as displayed in equation 2. So a vertex with a bigger weight gets a bigger share fraction of the score of surrounding vertices. That results in a much higher weighting for lonely higher rated words, for example single highlighted words, that are marked as important by the author.

$$S(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{w_i}{\sum_{V_k \in Out(V_j)} w_k} S(V_j) \tag{2}$$

C. Generating Tags for Images

TAG	IMG_URL	SCORE	WEIGHT	TYPE	COUNT	DOCLENGTH
garden	https://s-me...	83,918...	1	website	172	13.761
Vai	https://s-me...	64,117...	1	website	90	4.328
Jeep	https://s-me...	58,487...	1	website	92	3.326
vera	https://s-me...	52,302...	1	website	89	3.070
Wrang...	https://s-me...	51,796...	1	website	79	3.326
Long	https://s-me...	51,015...	1	website	87	845
weight	https://s-me...	48,996...	1	website	101	9.086
fat	https://s-me...	47,885...	1	website	81	6.197
fat	https://s-me...	47,885...	1	website	81	6.197
fat	https://s-me...	47,885...	1	website	81	6.197
Easter	https://s-me...	44,827...	1	website	59	2.974
essential	https://s-me...	43,574...	1	website	77	3.067
Marke...	https://s-me...	42,512...	1	website	55	5.131

Fig. 3. example for generated tags

To generate the image tags, we combine the textual content of a Pinterest pin with the content of the bookmarked page.

A pin contains the text of the pin description, the title of the container board, and the board's description (including repins). To generate tags, the system extracts the nouns and adjectives from the descriptions with the Stanford POS tagger [25]. Later on, the text rank algorithm ranks the nouns and adjectives based on their importance in the text and gives each word a score. We use the top 20 extracted terms from the pin's content as tags.

For the bookmarked page, we first need to download the actual content. Afterwards, we extract the *clutter*-free content of the webpage by using boiler pipe[5]. We use the *Article-Extractor*-class that uses shallow text features to retrieve the actual article text without any advertisements or navigational items. Here, we also apply the Stanford POS tagger, select the adjectives and nouns, and rank them according to the TextRank algorithm. Due to the varying text length of webpage it is necessary to adapt the tag selection. Therefore we introduce a threshold for the TextRank score. During our experiments a threshold of 1.0 shows promising results. Using this approach long document are able to get more tags assigned than short ones.

For the use of BM25 retrieval model the system additionally adds document length and term frequency of the top ranked tags to the analysis results. In Figure 3 an example set of tags and their corresponding fields is shown. The combination of tag and image url is unique in the database. We like to highlight that webpages can be referred by multiple pins resulting in multiple sources per image url.

D. Generating Search Query for Images from Full Text

Our system is tailored to support authors during the creation of an article. Our first prototype is shown in Figure 4, it simply contains of a WYSIWYG editor and a recommendation sidebar that contains the constantly updated image recommendation for the currently writing article. This author can then just copy and paste the image into the text. Therefore, we need to take full text as input for our image recommendation. This is contrary to image search systems like Google or Yahoo still retrieve relevant images based on keywords. We already introduced how to extract tags from the textual context of an image in Section IV-C.

Fig. 4. Supporting authors by recommending images while writing.

For this, the system extracts the same information like it does for the steps described . It extracts nouns and adverbs and then ranks the words with the Textrank algorithm. The top 20 scored words are then taken as input for the image retrieval system explained in section IV-E.

E. Image Retrieving

For the image retrieving process, two retrieval models are implemented - cosine similarity and BM25. Both systems use the same input data. The tags described in section IV-C and the search query tags from Section IV-D.

[5]https://code.google.com/p/boilerpipe/

1) Cosine Similarity: Cosine similarity is a vector space model that measures the angle between query vector and the image vector. Both vectors are n-dimensional vectors where each dimension is a value which represents a term. The image vector is build up from all tags extracted from the textual context and the value is the calculated TextRank score. One could also use the term frequency which has been calculated as well (refer to subsection IV-C). Since the importance of the tags are represented by the Textrank score, this measure promises better results. For the document the most important words are taken and their value is noted in the n-dimensional vector.

The cosine similarity measure assumes that the angle between two vectors represents how similar the documents, or in this case user text and image, are. Therefore, the images with similar tags to the important words of the user generated text are assumed to be relevant The formula for cosine similarity can be found in equation 3. Q is the vector of the users text, D is a vector representing an image and t is the dimension of the vectors.

$$s(Q,D) = \cos(Q,D) = \frac{\sum_{j=1}^{t} w_{qj} * w_{dj}}{\sqrt{\sum_{j=1}^{t} (w_{qj})^2 * \sum_{j=1}^{t} (w_{d_j})^2}} \tag{3}$$

To find the most relevant images, all images (with at least one common tag) are compared to the text vector. Than the top ranked images are returned to the user as shown in Figure 4.

2) BM25: BM25 is a probabilistic retrieval model which calculates the probability that a certain document is relevant to a query. To find relevant documents, BM25 calculates for each document (here image) a similarity score. For this it considers how often a query term (here, important tags of the user text) occurs in the document. Since a image has only unique tags, the term frequency of the tag in the external source or pin description are taken instead. Also, BM25 considers how often the important word occurs in the user generated text. Another factor is the frequency of the tag in the overall collection (e.g. document frequency). This inverse document factor improves the impact of words which occur infrequently and rates down the images which occur very often.

The formula for BM25 can be found in equation 4. Q is the query, here the relevant words of the users text. r_i is the number of relevant images having tag i. R is the number of relevant images. Relevance is not calculateable and can be only obtained from user feedback which is not yet implemented. n_i is the number of images tagged with i. N is the total number of images in the database. The k values are set empirically.

$$\sum_{i \in Q} \frac{(r_i + 0.5)/(R - r_i + 0.5)}{(n_i - r_i + 0.5)/N - n_i - R + r_i + 0.5} * \tag{4}$$

$$\frac{(k_1 + 1) * f_i}{K + f_i} * \frac{(k_2 + 1) * q * f_i}{k_2 + f_i} \tag{5}$$

The images which get the highest score are assumed to be the most relevant for the user and are returned to him.

V. Evaluation

To evaluate our approach we conducted multiple experiments. First, we run an evaluation using a manually constructed gold standard. The tested characteristic of the algorithm is whether it is able to return for a pin the appropriate tags. Second, we conduct a user study, allowing people to evaluate the quality of images returned by our system.

A. Goldstandard

To test the extraction of tags for a pin, we constructed a gold standard containing 110 pins. The pins are equally distributed over the eleven most active Pinterest categories. We randomly sub-sampled 10 pins per category. For every pin of every category, the top seven relevant tags were manually extracted and merged by two independent taggers. Therefore the gold-standard contains 770 manually extracted tags. The texts presented to the taggers to recommend tags are the pin itself (title, description, comments), the pin's board (title, description, category), repins of the pin and the external page (header, text) the pin refers to.

As in section IV-B explained, each tag has a score computed by the Textrank. For an evaluation of the tag relevance, the tags were ordered by their score. The evaluation algorithm compared the top 7 found tags for every pin in the gold standard with the corresponding manual selected ones.

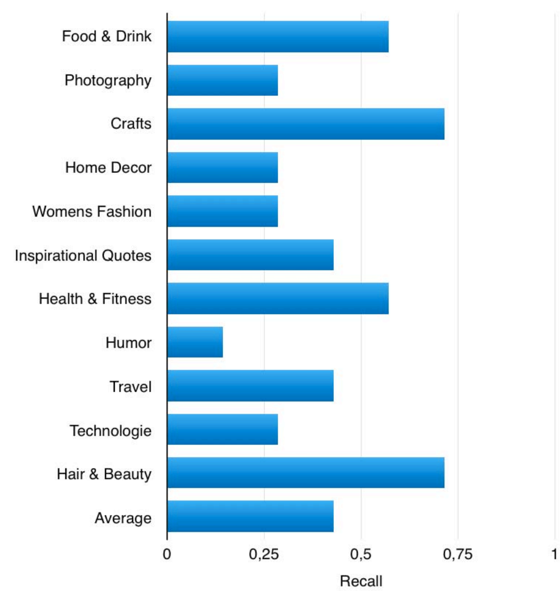

Fig. 5. The recall per category of matching tags.

The figure 5 shows the recall per category and the overall recall of $0,42$. On average four out of the seven manual derived tags were found. This is caused by the comparable higher weighting of tags extracted by the external webpage compared to the tag derived from the pin's textual context (description, board title, etc.). This can have two causes. On the one hand, the taggers might lay to much focus on the pin's content. On the other hand, it might be that the internal weight have to be adjusted. Nevertheless, the main task is to recommend images based on full text. Thus, a higher weight on the external pages will result in better retrieval performance. For future work it may be an option to damp the score of website tags or to boost the score of Pinterest description tags.

B. User Evaluation

For evaluating the quality of the search results, both the retrieval models BM25 and Cosine Similarity got tested as base retrieval model. Both of the retrieval models were extended for controlling the weighting of tags due to their sources. Therefore the system can balance the weighting between pin description, pin board and external website.

As first step, the retrieval of the original image was tested for the whole data set of 100000 pins. Given a pin, it is assumable that, using its description and the extracted text of the external site as search input, the pins original image is returned. For testing this assumption, the text of the external website was used as search input. In 96% of all cases the original image was retrieved as one of the top ten relevant images. In the remaining 4% the original image wasn't retrieved. The reason for that could be that neither the description of the pin nor the external site contained enough information. A common example for that is that links to external websites ending of a Top-Level-Domain tend to not contain the original image or only generic text. For example displays tumblr.com a description of the service itself and not the particular image the user wanted to pin.

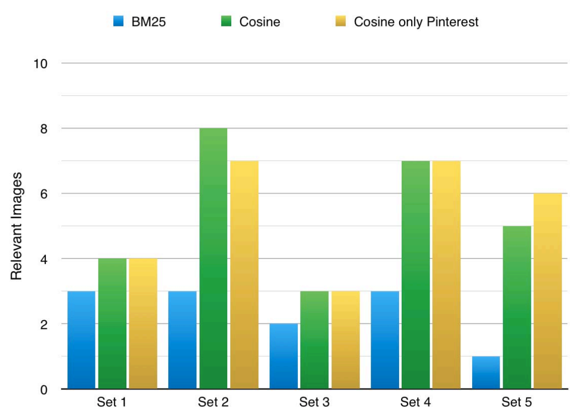

Fig. 6. The results of the user study.

Since this evaluation makes no statement about the quality of the retrieved images, a user study was done to get these information. Due to the purpose of image recommendation, users were asked, how many of the ten returned images are an appropriate representation for the given text. For this evaluation, five test sets were created by random sampling texts from the Pinterest data set. Every test set consists of one text retrieved from a pin's external webpage and six sets of images. Each of these sets contained ten images that were found by a retrieval model with a specific configuration. We tested for each retrieval model the number of relevant images with Pinterest-only context, external-only context, and both contexts. Twenty users participated in our study rating the images of each set.

Figure 6 shows the results of the user study for BM25 and cosine similarity with both context and cosine similarity only based on Pinterest. Each bar symbolizes the average number of appropriate images representing the given text.
The best results were achieved through Cosine Similarity. The use of Cosine Similarity resulted besides the original image in additionally more similar ones than BM25. Furthermore was the rating of most of the relevant images better than with BM25. Surprisingly Cosine Similarity without external sources gives approximately equivalent results as Cosine Similarity with external sources. Actually this configurations finds the image of the external website by only using information gained directly from Pinterest. Therefore it could be concluded that the pure information from Pinterest is enough to get relevant images for a given text.

C. Discussion

The result of BM25 using the external text only were 12.5% better than using information from Pinterest. This leads us to the conclusion that BM25 is not appropriate to match the small number of tags extracted from Pinterest. Therefore, cosine similarity is the favored method for our retrieval system.

By investigating the user study results we come along some edge case that lead to a performance decrease of our approach. For example images of "handmade smoke grenades" and "handmade cookies" end up in the same result set, because they share the tag "handmade". This is a common tag because the audience on Pinterest pins a big amount of images showing self-made things. Another problem occurs if images aren't described well enough. For example, if the external site is a Top-Level-Domain or contains a top x list describing other images besides the original one, it is most likely that the original image won't be found.

VI. Future Work

Despite the good results, there are some ideas how to improve the system. By using the Apache Nutch crawler as collection tool, we run into the problem of re-linking information. Nutch run a stepwise crawl cycle targeting multiple independent pages of a web host. Nevertheless, for gaining more insights into Pinterest it is of advantage to link more information together like author location and pin. Thus, we will implement a after-processing to find those links and present more findings.

Pinterest is not the only image based social network, our approach could be tested with other social networks like Flickr, Instagram or Tumblr to validate the findings. Especially, a social network that has a more distinct commentary culture could add much information that is helpful for tagging. That information could lead to even better results than we got from Pinterest. Another advantage of other social networks is the presence of an API which facilitates the process of getting a sufficient amount images. Although build a test set containing of a article text and an image is more difficult for the other networks.

Since the current version of the system is only a prototype, providing a Wordpress plugin would help user to seamlessly interact with the recommendation engine. Because of the full-text search, the application is especially suitable for helping writers of articles or blog entries to find a proper image for their purpose. The full-text approach causes that authors do not need to manually search for images they need but let the computer find out which parts of the article are the most relevant. By providing this functionality as a Wordpress plugin, the system could get relevance feedback from a huge user base, that could improve the results even more.

VII. CONCLUSION

In this paper, we presented an image recommendation approach based on the textual context extracted from Pinterest and the bookmarked pages. We focused on image recommendation task for full-text like blog posts or journalistic articles. Thereby, our recommendation engine can support author to find the appropriate images for their current work.

The textual context used consists of the pin description, the title of the container board, and the board's description (including repins). Further, we use the bookmarked pages of Pinterest as additional context. We presented the text processing chain consisting of POS tagging, lemmatization, and TextRank that is used to extract descriptive tags for images.

We compare two tag-based retrieval models for the image recommendation. We conclude that due to the small number of terms available cosine similarity performance better in comparison to BM25. The automatic validation shows that in 96% of the 100000 tested bookmarked pages the pin's image was in the top 10 results. Using consine similarity with the context of Pinterest gives up to 7 out of 10 relevant images recommended for a text during our user study.

REFERENCES

[1] D. Tam, "Facebook processes more than 500 tb of data daily," http://www.cnet.com/news/facebook-processes-more-than-500-tb-of-data-daily/, 2012, [Online; accessed 20-February-2015].

[2] R. K. Srihari, Z. Zhang, and A. Rao, "Intelligent indexing and semantic retrieval of multimodal documents," *Information Retrieval*, vol. 2, no. 2-3, pp. 245–275, 2000.

[3] K. Lerman, A. Plangprasopchok, and C. Wong, "Personalizing image search results on flickr," *Intelligent Information Personalization*, 2007.

[4] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst, "Faceted metadata for image search and browsing," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2003, pp. 401–408.

[5] S. Perez, "Pinterest appeals to publishers with new article pins," http://techcrunch.com/2013/09/24/pinterest-article-pins/, 2013, [Online; accessed 21-February-2015].

[6] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz, "Efficient and effective querying by image content," *Journal of intelligent information systems*, vol. 3, no. 3-4, pp. 231–262, 1994.

[7] C. W. Niblack, R. Barber, W. Equitz, M. D. Flickner, E. H. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin, "Qbic project: querying images by content, using color, texture, and shape," in *IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology*. International Society for Optics and Photonics, 1993, pp. 173–187.

[8] Y. Feng, J. Xiao, Y. Zhuang, and X. Liu, "Adaptive unsupervised multi-view feature selection for visual concept recognition," in *Computer Vision–ACCV 2012*. Springer, 2013, pp. 343–357.

[9] H. T. Shen, B. C. Ooi, and K.-L. Tan, "Giving meanings to www images." in *ACM Multimedia*, S. Ghandeharizadeh, S.-F. Chang, S. Fischer, J. A. Konstan, and K. Nahrstedt, Eds. ACM, 2000, pp. 39–47. [Online]. Available: http://dblp.uni-trier.de/db/conf/mm/mm2000.html#ShenOT00

[10] V. Harmandas, M. Sanderson, and M. D. Dunlop, "Image retrieval by hypertext links," in *ACM SIGIR Forum*, vol. 31, no. SI. ACM, 1997, pp. 296–303.

[11] W. X. Zhao, J. Jiang, J. He, Y. Song, P. Achananuparp, E.-P. Lim, and X. Li, "Topical keyphrase extraction from twitter," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, ser. HLT '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 379–388. [Online]. Available: http://dl.acm.org/citation.cfm?id=2002472.2002521

[12] C. Marlow, M. Naaman, D. Boyd, and M. Davis, "Ht06, tagging paper, taxonomy, flickr, academic article, to read," in *Proceedings of the seventeenth conference on Hypertext and hypermedia*. ACM, 2006, pp. 31–40.

[13] S. Robertson, H. Zaragoza, and M. Taylor, "Simple bm25 extension to multiple weighted fields," in *Proceedings of the thirteenth ACM international conference on Information and knowledge management*. ACM, 2004, pp. 42–49.

[14] E. Agichtein, E. Brill, and S. Dumais, "Improving web search ranking by incorporating user behavior information," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2006, pp. 19–26.

[15] C. Macdonald and I. Ounis, "Voting for candidates: adapting data fusion techniques for an expert search task," in *Proceedings of the 15th ACM international conference on Information and knowledge management*. ACM, 2006, pp. 387–396.

[16] M. Taylor, J. Guiver, S. Robertson, and T. Minka, "Softrank: optimizing non-smooth rank metrics," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*. ACM, 2008, pp. 77–86.

[17] M. Bilenko and R. W. White, "Mining the search trails of surfing crowds: identifying relevant websites from user activity," in *Proceedings of the 17th international conference on World Wide Web*. ACM, 2008, pp. 51–60.

[18] P.-N. Tan, M. Steinbach, V. Kumar *et al.*, *Introduction to data mining*. Pearson Addison Wesley Boston, 2006, vol. 1.

[19] S. Mittal, N. Gupta, P. Dewan, and P. Kumaraguru, "The pin-bang theory: Discovering the pinterest world," *arXiv preprint arXiv:1307.4952*, 2013.

[20] R. Ottoni, J. P. Pesce, D. B. Las Casas, G. Franciscani Jr, W. Meira Jr, P. Kumaraguru, and V. Almeida, "Ladies first: Analyzing gender roles and behaviors in pinterest." in *ICWSM*, 2013.

[21] A. Aizawa, "An information-theoretic perspective of tf–idf measures," *Information Processing & Management*, vol. 39, no. 1, pp. 45–65, 2003.

[22] R. Mihalcea and P. Tarau, "Textrank: Bringing order into texts." Association for Computational Linguistics, 2004.

[23] B. Tami Sutcliffe, "Exploring naming behavior in personal digital image collections: The iconology and language games of pinterest," Ph.D. dissertation, UNIVERSITY OF NORTH TEXAS, 2014.

[24] C. Hall and M. Zarro, "Social curation on the website pinterest. com," *proceedings of the American Society for Information Science and Technology*, vol. 49, no. 1, pp. 1–9, 2012.

[25] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*. Association for Computational Linguistics, 2003, pp. 173–180.