On the Use of Statistical Semantics for Metadata-based Social Image Retrieval

Navid Rekabsaz[1], Ralf Bierig[1], Bogdan Ionescu[2], Allan Hanbury[1], Mihai Lupu[1]

***Abstract*— We revisit text-based image retrieval for social media, exploring the opportunities offered by statistical semantics. We assess the performance and limitation of several complementary corpus-based semantic text similarity methods in combination with word representations. We compare results with state-of-the-art text search engines. Our deep learning-based semantic retrieval methods show a statistically significant improvement in comparison to a best practice Solr search engine, at the expense of a significant increase in processing time. We provide a solution for reducing the semantic processing time up to 48% compared to the standard approach, while achieving the same performance.**

I. Introduction

When referring to *semantics* or *concepts*, the image and text Information Retrieval communities have slightly different expectations. In the image retrieval community *semantics* generally refers to a transformation from vectors (bitmaps) to natural language terms. Since the focus of this paper is on the retrieval of social images using textual contexts, we refer to *semantics* as a transformation from terms to a vector in a vector space where proximity is indicative of conceptual similarity. To achieve these goals, some make use of knowledge databases e.g., WordNet, OpenCyc. These databases determine word meanings from a priori and explicitly human-entered knowledge. However, such methods cannot be applied to less supported languages or to more specific domains without pre-existing knowledge.

In this paper, we focus on text retrieval in the particular context of social retrieval where data is accompanied by user generated text metadata. We propose a comparative study of the effect of statistical semantics on metadata text for the purpose of image retrieval. To the best of our knowledge, this is the first such study and we have managed to show that, given an appropriate similarity function, such methods can bring a valuable and significant contribution to image retrieval. The study also shows that, we can substantially improve the processing time, without loss of performance.

A. Motivation

Text similarity measures are still important to image retrieval. Despite the recent success of image deep learning methods [8], text remains an important source of information in many cases where tags, descriptions, or other textual contexts are associated with the image. Nevertheless, and understandably, the focus resided on image processing and, so far, the methods used for text similarity for the purpose of image retrieval are fairly mainstream [17]. Such methods fail when two texts use a disjunct vocabulary to describe the same fact or situation. Despite a generally strong interest on word-to-word or sentence-to-sentence similarity [5], research on image metadata document retrieval level is limited.

There is in fact a lack of understanding of how statistical semantics methods perform on this type of documents. While it is known that statistical semantic methods rely on the tendency of natural language to use semantically related terms in similar contexts, it is unclear if they would still work on documents consisting of image tags.

Therefore, in this study we address two questions:

1) Can statistical semantics outperform state-of-the-art text search engines in terms of text-based image retrieval effectiveness?
2) Among statistical semantics methods, do newer approaches (i.e., deep learning) outperform older ones (i.e., random indexing)?

The remainder of this section provides some background and related work. Then, Section II describes the similarity models explored to answer the two questions above, followed by the outline of our experiments in Section III. Results are discussed in Section IV.

B. Background and Related Work

Textual features has been used in many multimodal retrieval systems. For instance, recently, Eskevich et al. [4] considered a wide range of text retrieval methods in the context of multimodal search for medical data, while Sabetghadam et al. [15] used text features in a graph-based model to retrieve images from Wikipedia. However, these works do not exploit in particular text semantics.

In the text retrieval community, text semantics started with Latent Semantic Analysis/Indexing (LSA/LSI) [3], the pioneer approach that initiated a new trend in surface text analysis. LSA was used for image retrieval [13], but the method's practicality is limited by efficiency and scalability issues caused by the high-dimensional matrices it operates on. Explicit Semantic Analysis (ESA) is one of the early alternatives, aimed at reducing the computational load [9]. However, unlike LSA, ESA does rely on a pre-existing set of concepts, which may not always be available. Random Indexing (RI) [16] is another alternative to LSA/LSI that creates context vectors based on the occurrence of words contexts. It has the benefit of being incremental and operating

*This work has been supported by MUCKE project (FWF number 213457)

[1] Information and Software Engineering Group, Vienna University of Technology, A-1040 Vienna, Austria `[last name]@ifs.tuwien.ac.at`

[2] LAPI, University Politehnica of Bucharest, 061071 Romania `bionescu@alpha.imag.pub.ro`

with significantly less resources while producing similar inductive results as LSA/LSI and not relying on any pre-existing knowledge. Word2Vec [12] further expands this approach while being highly incremental and scalable. When trained on large datasets, it is also possible to capture many linguistic subtleties (e.g., similar relation between Italy and Rome in comparison to France and Paris) that allow basic arithmetic operations within the model. This, in principle, allows exploiting the implicit knowledge within corpora. All of these methods represent the words in vector spaces.

Approaching the text semantics, Liu et al. [10] introduced the Histogram for Textual Concepts (HTC) method to map tags to a concept dictionary. However, the method is reminiscent of ESA described above, and it was never evaluated for the purpose of text-based image retrieval.

Regardless of the semantic representation (all of them are vectors), a vital aspect of retrieval is the similarity function.

II. Similarity Methods

Similarity between two documents based on the semantics of their terms can be approached in two ways: 1. aggregating the terms' vectors in a document vector and computing similarity between two such vectors, or 2. identifying similar terms between documents and aggregating the similarity values of pairs of terms in the two documents.

We denote the first approach $SimAgg$, defined in Eq. 1:

$$V_A = \sum_{i=1}^{n} idf_i * A_i \qquad SimAgg(A,B) = Cos(V_A, V_B) \tag{1}$$

where V_A represents the vector representation of document A, A_i is the vector representation of the ith word, n is the number of words in the document, idf_i is the Inverse Document Frequency of the ith word in the corpus and Cos() denotes the cosine distance. The method creates a representation vector for each document by aggregating the vectors of the words in the document. We define the aggregation method as the weighted sum of the elements of the word vectors. Having the document vectors, we calculate the similarity with the traditional cosine function.

The second approach, denoted $SimGreedy$ [11], is based on $SimGreedy(A, B)$:

$$SimGreedy(A,B) = \frac{\sum_{i=1}^{n} idf_i * maxSim(A_i, B)}{\sum_{i=1}^{n} idf_i} \tag{2}$$

where the function $maxSim$ calculates separately the cosine of the word A_i to each word in document B and returns the highest value. In this method, each word in the source document is aligned to the word in the target document to which it has the highest semantic similarity. Then, the results are aggregated based on the weight of each word to achieve the document-to-document similarity. $SimGreedy$ is defined as the average of SimGreedy(A,B) and SimGreedy(B,A).

It should be noted that Rus et al. [14] expand the method with a penalizing factor to remove low similarities as noise. We found that for the particular case of social image retrieval this factor was ineffective. In fact, instead of filtering noise, it tends to reduce all values evenly without any re-ranking benefit. Therefore, these results are not reported here.

The time complexities of the two methods are very different. If n and m are the number of words in document A and B respectively, the complexity of SimAgg is of order $n+m$ while SimGreedy is of order $n*m$.

III. Experiments

The evaluation was conducted using Flickr data, in particular in the framework of the MediaEval Retrieving Diverse Social Images Task 2013/2014 [6], [7]. The task addresses result relevance and diversification in social image retrieval. We merged the datasets of 2013 (Div400) [7] and 2014 (Div150Cred) [6] and denoted it as MediaEval. It consists of about 110k photos of 600 world landmark locations (e.g., museums, monuments, churches, etc.). Location information include a ranked list of photos, a representative text, Flickr's metadata, a Wikipedia article of the location and user tagging credibility estimation (only for 2014 edition). For semantic text similarity, we focus on the relevance of the representative text of the photos containing title, description and tags. We removed HTML tags and decompounded the terms using a dictionary obtained from the whole corpus. In addition to the similarity methods defined in Section II, we also considered the asymmetric version of SimGreedy (SimGreedy(Q,D) and SimGreedy(D,Q)).

We used the English Wikipedia text corpus to train our word representation vectors based on Word2Vec and Random Indexing, each with 200 and 600 dimensions. We cleaned the corpus by removing HTML tags and non-alphabetic characters. We trained our Word2Vec word representation using Word2Vec toolkit[1] by applying CBOW approach of Mikolov et al. [12] with context windows of 5 words and subsampling at $t = 1e^{-5}$. The Random Indexing word representations were trained using Semantic Vectors package[2]. We used the default parameter settings of the package which considers whole the document as context window. In both Word2Vec and Random Indexing we considered the words with frequency less than five as noise and filtered them out.

Additionally, as a sanity check on the ability of the proposed representations and similarity methods, we tested them on SemEval 2014 Multilingual Semantic Textual Similarity - Task 10 [1], the English subtask. The goal of this task is to measure the semantic similarity of two sentences. Participating systems are compared by their mean Pearson correlation between the system output and a human-annotated gold standard. A good result in this tasks would confirm the ability of the methods for identifying semantic similarity in general texts, and allow us to observe eventual differences imposed by the specificity of the image metadata text genre.

IV. Results and Discussion

As mentioned above, we first check the sanity of the methods and word representations on SemEval 2014 Task 10 [1]. Table I shows the Mean Pearson correlations between

[1] https://code.google.com/p/word2vec/

[2] https://code.google.com/p/semanticvectors/

TABLE I

Mean Pearson correlation of SemEval 2014 Task 10 [1] using Word2Vec (W2V) [12] and and Random Indexing (RI) [16] word representation

Representation	Dim	SimAgg	SimGreedy
RI	600	0.691	0.706
RI	200	0.678	0.702
W2V	600	0.685	**0.715**
W2V	200	0.654	0.715

TABLE II

Models trained on Wikipedia. (Q,D) and (D,Q) are SimGreedy(Q,D) and SimGreedy(D,Q). Digits in brackets indicate the ID of each run. † denotes statistical significant difference in comparison to the Solr baseline

Repr.	Dim	SimAgg	SimGreedy	(Q,D)	(D,Q)
RI	200	0.774 (1)	†0.788 (5)	0.704	0.766
RI	600	0.766 (2)	†0.787 (6)	0.703	0.769
W2V	200	0.778 (3)	†**0.795** (7)	0.690	0.760
W2V	600	0.779 (4)	†0.793 (8)	0.693	0.757

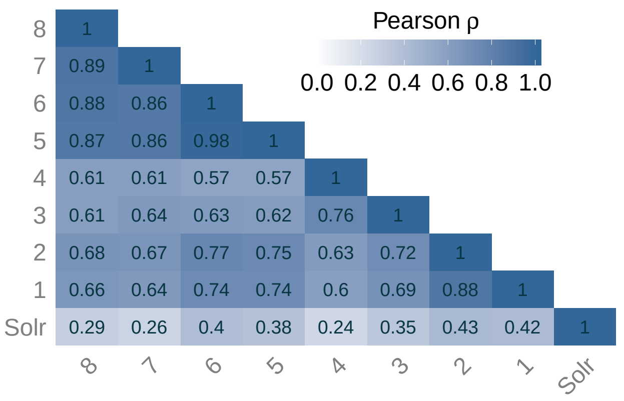

Fig. 1. Pearson Correlations between all 8 combinations of approaches and the Solr baseline. The numbers refer to the ID of the runs in Table 2

TABLE III

Models trained on MediaEval corpus. † denotes statistical significant difference in comparison to the Solr baseline.

Representation	Dim	SimAgg	SimGreedy
RI	200	†**0.795**	0.776
W2V	200	0.767	0.758
RI-Window	200	0.753	0.759

the similarity methods and the gold standard. The most impressive result is that SimGreedy with Word2Vec achieved an average correlation of 0.71 as the best overall performance. This represents rank 11th out of the 38 submitted runs. However, all 10 runs above use a knowledge base and/or NLP which would not generalize to other domains or languages. Between similarity methods, SimGreedy shows better performance than SimAgg. It also appears that the similarity method has more effect on the results than the number of dimensions or word representation.

In the following, we focus on MediaEval Retrieving Diverse Social Images Task 2013/2014 [7], [6] considering the evaluation metric as the precision at a cutoff of 20 documents (P@20) which was also used in the official runs. A standard Solr index was used as the baseline. It produced a P@20 of 0.76. For all experiments, statistical significance against the baseline was calculated using Fisher's two-sided paired randomization test. In all tables, † denotes statistical significant difference at p 0.05 or lower.

The results of evaluating the combination of methods and word representations are shown in Table II. We see that SimGreedy outperforms SimAgg regardless of the training method while all runs with SimAgg have no significant difference from the baseline. Observing the asymetric versions of SimGreedy, SimGreedy(D,Q) shows better results than SimGreedy(Q,D) since documents are generally longer and more descriptive than queries. However SimGreedy outperformed both SimGreedy(Q,D) and SimGreedy(D,Q). We hypothesize that the (Q,D) version, which performs very poorly on its own, acts as a length normalization factor for the (D,Q) version, therefore contributing to the improved result.

For more insight on the differences between the runs, we additionally compared all our combinations by calculating their pairwise Pearson rank correlation (Figure 1). The average correlation between runs using SimGreedy is larger than those using SimAgg. This means that regardless of the training method and the number of dimensions for word representation, using SimGreedy produces more similar results. We also observe very high correlations between the same models with 200 and with 600 dimensions which demonstrates that increasing the dimensionality does not affect results.

While Wikipedia provides a general knowledge about different topics, it is interesting to study the effectiveness of using the MediaEval corpus for representing the words instead of an external resource. Therefore, we train the Word2Vec and Random Indexing models on the MediaEval corpus. As the previous experiments show the ineffectiveness of dimensionality, we trained the models only with 200 dimensions.

The results in Table III show an overall lower performance. Exceptionally, we observe a significantly better performance when using Random Indexing in combination with SimAgg - as good as the best result achieved by the previous experiment. As the default definition of Random Indexing uses the whole document as the context window, while Word2Vec uses a context window of 5, we also trained Random Indexing with the context window of 5 (RI-Window). As it is shown in Table III, similar to Word2Vec, RI-Window does not improve the performance. It is therefore reasonable to assume that the difference is due to the amount of information considered in creating the vector of each term.

This observation leads to the hypothesis that, as additional information proved useful in the construction of the terms, it should also prove useful in the construction of the queries themselves. Therefore, in the following, we expanded the

TABLE IV

RESULTS USING QUERY EXPANSION. † DENOTES STATISTICAL SIGNIFICANT DIFFERENCE IN COMPARISON TO THE SOLR BASELINE

Corpus	Repres.	Dim	SimAgg	SimGreedy
Wiki	RI	200	0.768	†0.794
Wiki	W2V	200	0.756	†0.786
MediaEval	RI	200	†**0.795**	†0.788
MediaEval	W2V	200	†0.780	†0.792

TABLE V

MEDIAEVAL2014 RESULTS USING QUERY EXP.

Corpus	Repres.	Dim	SimAgg	SimGreedy
Wiki	RI	200	0.795	0.833
Wiki	W2V	200	0.788	0.813
MediaEval	RI	200	0.84	0.82
MediaEval	W2V	200	0.831	**0.848**

topic names with the first sentence of their corresponding Wikipedia page. As it is shown in Table IV, in comparison to previous experiments, the performance did not change significantly except the results related to SimGreedy in combination to MediaEval corpus. We concluded that using SimGreedy beside query expansion provides a stable method with good performance regardless of the word representation method or training corpus.

In order to compare the results with the participating systems in the task, we repeated the experiment on test dataset 2014. As it is shown in Table V, using SimGreedy and Word2Vec trained on the MediaEval corpus, we achieved the state-of-the-art result of 0.848 for P@20 between 41 runs including even the ones which used image features but not external resources.

Although SimGreedy shows stable and better performance in comparison to SimAgg, based on the time complexity discussion in Section II, it has a much longer execution time. We observed that SimGreedy is approximately 40 times slower than SimAgg so that SimGreedy generally has an execution time of about 25 to 30 minutes while it takes less than a minute for SimAgg. We therefore turned the procedure into a two-phase process [2]. In the first phase, we applied the SimAgg method to obtain a first ranking of the results. As the second phase, we used n percent of the top documents ranked by the first phase and re-ranked them using SimGreedy. For each combination of different parameters (training data, dimensionality, training method) and for all the values of n from the 1 to 100, we found an extremely similar behaviour summarized in Figure 2. In order to find the best value for n as the cutting point, we identified the highest precision value that is not significantly different from the best one (i.e. when n is 100 percent). This corresponds to $n = 49$. Giving the second phase (SimGreedy) is about 40 times slower than the first (SimAgg), using this approach reduces the execution time to 48 percent while the performance remains the same.

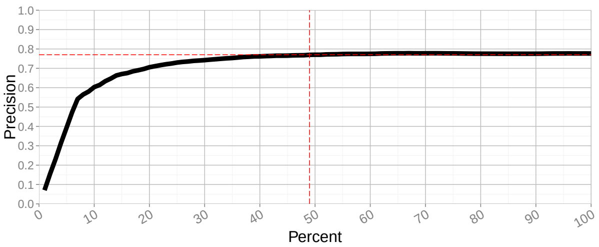

Fig. 2. Average performance of the two-phase approach with best value at around 49%

V. CONCLUSIONS AND FUTURE WORK

We explored the use of semantic similarity in text-based image retrieval in social media domain by applying Word2Vec and Random Indexing together with two similarity methods. We ran experiments on the SemEval2014 Task 10 and the MediaEval Retrieving Diverse Social Images Task 2013/2014. Beside achieving state-of-the-art results on both datasets, we show that the similarity method has more effect on the results rather than the number of dimensions or word representation training method. In addition, by using a two-phase approach, we reduced in half the processing time of the best run while keeping precision within the boundary of statistically insignificant difference.

REFERENCES

[1] E. Agirrea, C. Baneab, C. Cardiec, D. Cerd, M. Diabe, A. Gonzalez-Agirrea, W. Guof, R. Mihalceab, G. Rigaua, and J. Wiebeg. Semeval-2014 task 10: Multilingual semantic textual similarity. *SemEval*, 2014.

[2] V. Dang, M. Bendersky, and W. Croft. Two-stage learning to rank for information retrieval. In *Proc. of ECIR*, 2013.

[3] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *JASIS*, 1990.

[4] M. Eskevich, G. J. Jones, R. Aly, and et al. Multimedia information seeking through search and hyperlinking. In *Proc. of ICMR*, 2013.

[5] L. Han, A. Kashyap, T. Finin, J. Mayfield, and J. Weese. UMBC EBIQUITY-CORE: Semantic textual similarity systems. In *Proc. of *SEM*, 2013.

[6] B. Ionescu, A. Popescu, M. Lupu, A. L. Gînsca, B. Boteanu, and H. Müller. Div150cred: A social image retrieval result diversification with user tagging credibility dataset. *ACM Multimedia Systems Conference Series*, 2015.

[7] B. Ionescu, A.-L. Radu, M. Menéndez, H. Müller, A. Popescu, and B. Loni. Div400: a social image retrieval result diversification dataset. In *Proc. of ACM Multimedia Systems Conference Series*, 2014.

[8] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[9] C. Liu and Y.-M. Wang. On the connections between explicit semantic analysis and latent semantic analysis. In *Proc. of CIKM*, New York, NY, USA, 2012.

[10] N. Liu, E. Dellandréa, L. Chen, C. Zhu, Y. Zhang, C.-E. Bichot, S. Bres, and B. Tellez. Multimodal recognition of visual concepts using histograms of textual concepts and selective weighted late fusion scheme. *CVIU*, 2013.

[11] R. Mihalcea, C. Corley, and C. Strapparava. Corpus-based and knowledge-based measures of text semantic similarity. In *Proc. of AAAI*, 2006.

[12] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[13] T.-T. Pham, N. Maillot, J.-H. Lim, and J.-P. Chevallet. Latent Semantic Fusion Model for Image Retrieval and Annotation. In *Proc. of CIKM*, 2007.

[14] V. Rus, M. Lintean, R. Banjade, N. Niraula, and D. Stefanescu. SEMILAR: The Semantic Similarity Toolkit. In *Proc. of ACL*, 2013.

[15] S. Sabetghadam, M. Lupu, R. Bierig, and A. Rauber:. A combined approach of structured and non-structured ir in multimodal domain. In *Proc. of ICMR*, 2014.

[16] M. Sahlgren. An introduction to random indexing. In *Methods and Applications of Semantic Indexing Workshop in the Proc. of TKE*, 2005.

[17] B. Thomee and A. Popescu. Overview of the ImageCLEF 2012 Flickr Photo Annotation and Retrieval Task. In *Proc. of CLEF*, 2012.