Graph Retrieval with the Suffix Tree Model

Mathias Lux[1] and **Sven Meyer zu Eissen**[2] and **Michael Granitzer**[3]

Abstract. The paper in hand presents an adoption of the suffix tree model for the retrieval of labeled graphs. The suffix tree model encodes path information of graphs in an efficient way and so reduces the size of the data structures compared to path index based approaches, while offering a better runtime performance than subgraph isomorphism based methods. Within a specific use case we evaluate the correlation of the developed method to human judgement and compare the correlation values to other methods. We show that in our use case, which is the retrieval of digital photos annotated with MPEG-7 using the *MPEG-7 Semantic Description Scheme,* the presented algorithm performs better than other methods.

1 INTRODUCTION

Let $G = \langle V, E \rangle$ be a graph, where V denotes the node set and $E \subseteq V \times V$ denotes the edge set. Given a query graph G_q and a graph set $\mathcal{G}$, graph retrieval deals with the task to identify a subset $\mathcal{R} \subseteq \mathcal{G}$ with the property

$$\forall G \in \mathcal{R} : \varphi(G_q, G) \geq t$$

where $\varphi : \mathcal{G} \times \mathcal{G} \rightarrow \mathbf{R}$ denotes a similarity function and $t \in \mathbf{R}$ is a minimum similarity threshold.

The research question how to search similar graphs in a database was already prescribed in a work by Simmons in 1966 (see [13]), in which he matched conceptual graphs. Since then, different applications areas emerged; they include querying chemical graph databases that store molecular structures, retrieving vector and raster images using characteristics encoded in a graph, and recently, searching in semantically enriched data in the context of semantic Web applications.

Our application scenario relates to multimedia retrieval with the MPEG-7 standard, where metadata are represented as graphs: A user formulates his or her information need in the form of a graph, which is then matched against an MPEG-7 graph database $\mathcal{G}$.

A property of MPEG-7 graphs is that their nodes and edges are labeled with text, say, for each $G \in \mathcal{G}$ there exists a function $l_E : E \rightarrow T_E$ as well as $l_V : V \rightarrow T_V$, where T_E, T_V are term sets. The goal is to retrieve graphs that match both, the query graph's structure as well as the labels. The challenges in this connection are twofold:

1. The statement of a similarity function φ that reflects the application scenario, and
2. The operationalization of the retrieval functionality.

The second challenge restricts the flexibility in formulating a similarity function: φ must not be expensive to evaluate in terms of runtime complexity since in our case a user waits actively for retrieval results.

[1] University of Technology Graz, Knowledge Management Institute, Austria, email: mathias.lux@tugraz.at
[2] Bauhaus University Weimar, Germany, email: sven.meyer-zu-eissen@medien.uni-weimar.de
[3] Know-Center Graz, Austria, email: mgrani@know-center.at

2 RELATED WORK

Although maximum common subgraph isomorphism is a natural starting point for graph similarity computation (see [3]), it cannot be applied to our scenario: First, the question if two graphs G and H contain an isomorphic subgraph whose edge set has more than $k \in \mathbf{N}$ elements is NP-complete (see [6]). Second, quantifying similarity using ratios of subgraph edge set sizes solely may not reflect our problem, since edge label matches can be of different importance, depending on the value of an edge label.

For this and similar reasons, graph retrieval algorithms are tailored to the requirements of the underlying use case. For example, Fonseca et al. used graph invariants of trees—in this specific case the eigenvalues of the tree's and subtree's adjacency matrix—to identify relevant cliparts represented as trees, representing adjacency and inclusion of color areas within the cliparts, in a database (see [5],[12]).

Zong et al. (see [18]) retrieved labeled graphs using an index in which the labels of paths up to a certain length were stored. The relevance between a query graph and a graph from the database was computed from a TF*IDF-like similarity measure that was applied to the edge labels.

Berreti et al. (see [2]) extracted information on neighbouring colour regions from raster images, which was encoded in directed labeled graphs. To retrieve similar images a graph database was queried employing a tailored metric, which proved as slow but highly configurable.

2.1 Contribution

Text retrieval methods based on the vector space model, especially those using inverted lists as described in [1], have been applied to graph retrieval before: A graph's labels form a virtual document; likewise, the query graph's labels are used to construct a query document. The similarity between these documents is computed using the vector space model along with standard similarity measures like TF*IDF or BM-25.

Unlike traditional vector space approaches our proposed method employs the suffix tree model, described in [8]. Its advantage is that similarity computations incorporate word order within sentences and text fragments. Applied to the outlined MPEG-7 retrieval scenario, this property is especially

useful when matching labels in a graph's paths, yielding to better similarity values like the respective experiments show.

3 APPLICATION SCENARIOS

The specification of semantics often follows a graph modeling approach; the pioneering work of Sowa (see [14]) is one of many examples. Similarity search in this and related contexts reduces to graph retrieval.

Currently a trend towards a semantically enriched Web can be noted. This movement started with the vision of a semantic Web by Berners-Lee (see e.g. foreword in [4]) and resulted in the definition of a syntax for semantics, formally defined in an ontology language based on the Resource Description Framework (RDF), which uses a model based on directed labeled graphs.

Another initiative, aimed at an interoperable standards for multimedia data, is the Moving Picture Expert Group, in short MPEG. Within their Multimedia Content Description Interface, short name MPEG-7, they defined a way to semantically describe the contents of multimedia files by interconnecting semantic objects (e.g. agents, places, and so on) by typed semantic relations (see [7] for more details), which again results in directed labeled graphs that encode semantics.

All of the above mentioned scenarios model semantics with directed labeled graphs. While the same edge label can be used more than once within a graph, we assume that node labels are unique within a graph as defined in MPEG-7, RDF and conceptual graphs.

4 APPLYING THE SUFFIX TREE MODEL TO GRAPH RETRIEVAL

Information retrieval methods that have been used in the past for graph retrieval have in common that they transform database graphs $G_i \in \mathcal{G}$ as well as query graphs G_q to documents d_i and d_q, respectively, which are then compared using their vector space model representations in combination with a related similarity measure like the cosine similarity. Here, the documents consist of sentences, which are made up of node and edge label concatenations from paths in the corresponding graphs. This methodology raises two questions:

1. Which paths of a graph should be used for the construction of d_i and d_q?
2. Which retrieval methodology should be chosen for query matching?

With respect to point (1), some heuristics have been proposed. One prominent method is discussed in connection with *GraphGrep* (see [11]). The paths of a graph are extracted either by identifying all paths in a graph up to a certain length, e.g. with a depth first or breadth first search starting from each vertex (see e.g. [15]), or by identifying frequent substructures within the graphs (see e.g. [17] or [16]).

The focus of our research refers to point (2). Known graph retrieval methods that rely on the vector space model disregard term order or include only partial term order information when using n-grams for indexing. In the following, a similarity measure is presented that tackles the aforementioned problem; it compiles *full* path label order information into the similarity values while keeping the computational complexity bounded by a linear function. In this connection, knowledge about suffix trees is necessary prerequisite; some details are summarized in the next section.

4.1 Suffix Trees

The ith suffix of a document $d = w_1 \dots w_m$ is the substring of d that starts with word w_i. A suffix tree of d is a labeled tree that contains each suffix of d along a path whose edges are labeled with the respective words. The construction of a suffix tree is straightforward: The ith suffix of d is inserted by checking whether some edge emanating from the root node is labeled with w_i. If so, this edge is traversed and it is checked whether some edge of the successor node is labeled with w_{i+1}, and so on. If, in some depth k, a node n without a matching edge is reached, a new node is created and linked to node n with an edge labeled with w_{i+k}.

Figure 1 illustrates a the suffix tree in which the documents d_1 ="Boy plays chess" and d_2 ="Boy plays bridge too" have been inserted.

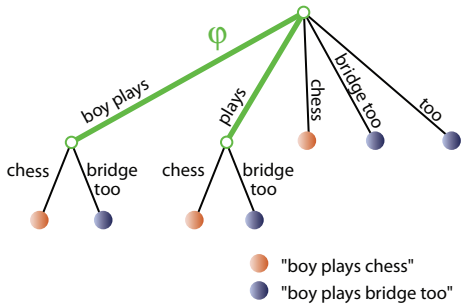

Figure 1. A suffix tree in which the documents d_1 ="Boy plays chess" and d_2 ="Boy plays bridge too" have been inserted.

4.2 Path-based Graph Suffix Trees

Let d_i denote the document that is associated with G_i, and likewise, let d_q denote the document that is associated with G_q. Both, d_i and d_q consist of "sentences", which are concatenations of path labels from selected paths from G_i and G_q, following a heuristic mentioned above.

A natural similarity measure between d_i and d_q arises when inserting each suffix from each sentence of d_i and d_q into an initially empty suffix tree $G_S = \langle V_S, E_S \rangle$. Let $E_i \subseteq E_S$ denote the set of the edges that have been traversed when all suffixes of d_i's sentences have been inserted into G_S, and analogously, let $E_q \subseteq E_S$ denote the traversed edge set for all sentences' suffixes from d_q. The similarity between d_i and d_q can be measured by how many edges E_i and E_q have in common, e.g. quantified by the Jaccard coefficient:

$$\varphi_S(G_i, G_q) = \frac{\mid E_i \cap E_q \mid}{\mid E_i \cup E_q \mid}$$

Furthermore in [8] two more weighting schemes using term frequency and inverse document frequency of edges, are described to enhance relevance and precision. For similarity calculation of graphs such a weighting can be applied.

In addition to the two original weighting schemes a third scheme relying solely on IDF can be introduced. Stripping the term frequency from the original weighting formula, a similarity measure can be defined as follows:

$$\varphi_{idf}(G_i, G_q) = \frac{1}{\mid E_S \mid} \sum_{e \in E_S} traversed(e) \cdot IDF(e)$$

$$\text{with } traversed(e) = \begin{cases} 0 & e \notin E_i \cap E_q \\ 1 & e \in E_i \cap E_q \end{cases}$$

Here, $IDF : E \rightarrow \mathbf{R}$ is defined to be the inverse document frequency function, $IDF(e) = \log(\frac{n}{S(e)})$, with n being the total number of documents and $S : E \rightarrow \mathbf{N}$ denoting the function that delivers the number of distinct documents that traversed a given edge on insertion into the suffix tree.

5 EVALUATION

Although the presented suffix tree model for graphs can be applied to arbitrary graphs with node and edge labels, the evaluation was done within a multimedia retrieval scenario: Using MPEG-7, the *Multimedia Content Description Interface*, multimedia documents can be annotated using graphs expressing the semantics of the multimedia document. This particular functionality of MPEG-7 is defined in the Semantic Description Scheme (see [7] for details on MPEG-7).

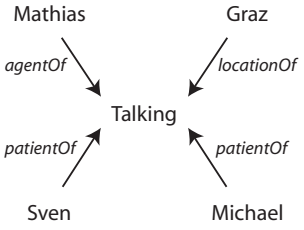

Figure 2. Illustration of an MPEG-7 based annotation expressing that *Mathias is talking to Sven and Michael in Graz.*

Within this scenario two graphs, like the one shown in figure 2, can be compared and a similarity value can be obtained. Based on the used mechanism for similarity calculation different results are achieved. Our evaluation aims to identify the most semantic method (in terms of human judgement) for similarity calculation of MPEG-7 based annotations.

To evaluate the *semantics* of candidate similarity measure a test set of 96 manually annotated digital photos was used. In essence for all photos a labeled directed graph exists, which describes the semantics of the image by specifying persons, time points, locations and events as nodes and interconnecting these nodes by labeled edges, like shown in figure 2. The graphs have a median number of nodes of 5.81, with a medium number of 5.99 edges. From this test data set 20 photo pairs were identified, which were used to create a questionnaire. The participants of the evaluation were asked to rate the pairwise similarity of the photos. The averaged similarity from the participants answers was correlated to the results of the candidate similarity measures.

After initial evaluations of 18 and 15 participants a final evaluation with 112 participants was carried out. The results

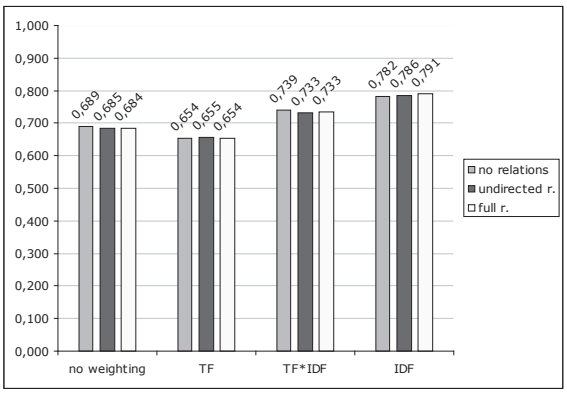

Figure 3. Evaluation of the Suffix Tree Metric in correlation to human judgement

of the evaluation of the suffix tree model based metrics is shown in figure 3. With each weighting scheme three different strategies for building the tree are evaluated: A first approach is to build the tree without taking the edge labels into account (shown as option *no relations* in figure 3), so only the sequence of node labels is inserted into the tree. A second approach is to *normalize* all relation labels without taking their directions into account (shown as option *undirected r.* in figure 3). This can be done by ignoring all direction information on edges. The third option is to use the full paths including node and edge labels (shown as option *full r.* in figure 3).

As can be seen easily the suffix tree model cannot provide an optimal approximation of human judgement with any of the presented weighting schemes. With no weighting schema a rounded maximum correlation value of 0.689 can be achieved. With the term frequency weighting, which was proposed in the original publications the correlation value even gets worse. The inverse document frequency (IDF) weighting proposed in this publication offers the best correlation with a maximum value of 0.791 taking all node and edge information (labels and direction) into account.

Besides the above introduced suffix tree similarity measure for graphs following similarity measures from text and graph retrieval were compared to human judgement:

1. Vector space based on node and edge labels, cosine coefficient as similarity measure with following weighting schemes. This metric does not take the structure of the graph into account, the set of labels is treated as text document:
   (a) without weighting scheme (*Text VS* in fig. 4)
   (b) TF*IDF (*Text VS TF*IDF* in fig. 4)
   (c) BM25 (*Text VS BM25* in fig. 4, see [9] and [10] for details on BM25)
2. Vector space with graph paths as terms, cosine coefficient as similarity measure with following weighting schemes:
   (a) TF*IDF on paths with one arc (*VS IDF Triple* in fig. 4) and full length paths (*VS IDF Paths* in fig. 4)
   (b) BM25 on paths with one arc (*VS BM25 Triple* in fig. 4) and full length paths (*VS BM25 Paths* in fig. 4)
3. Maximum common subgraph metric from [3] (*MCS* in fig.

4)

4. Error correcting subgraph isomorphism metric from [2] with boolean edge label distance functions and two options for used node label distance functions:
   (a) Boolean distance function (*Berretti (Bool)* in fig. 4)
   (b) Term vector distance function (*Berretti (VS)* in fig. 4)

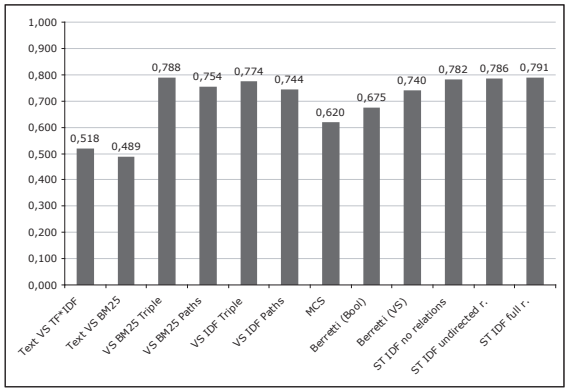

Figure 4. Evaluation of different distance functions and metrics using the correlation to human judgement

The evaluation results in figure 4 show that the suffix tree model with proposed inverse document frequency weighting offers the best correlation to human judgement in the presented domain. However the *VS BM25 Triple* metric offers a nearly as high correlation value. The two variants of the error correcting subgraph isomorphism metric of [2] do not perform as good as the other candidates. All evaluated text based similarity and distance measures, which do not take the structure in to account, do not correlate well with human judgement.

6 CONCLUSION

As can be seen easily from the evaluation similarity measures, which take the structure information of the graphs into account, are superior to the tested text retrieval mechanisms, which use node and edge labels for retrieval. The suffix tree method has a slightly better correlation coefficient and therefore reflects human judgement better than the other methods. However the difference to the vector space method is marginal, which justifies for example the usage of an path index for graph retrieval. One possible explanation why the triple based VS approach performs that good is that in the inspected domain all node labels are unique within a single graph.

The most interesting point is, that methods adapted from text retrieval perform better than the evaluated methods developed for graphs, like MCS and the algorithm of Berretti et al. described in [2] on the used test data set. However the number of photos in the set is too small for general conclusions, but as no test data sets for semantic annotations currently exist, the creation of semantic annotations for multimedia documents is a laborous task and the usefulness of random graphs for evaluation is limited in this domain, an evaluation with a bigger data set was out of scope of the project. Nevertheless the presented evaluation provides a starting point for further investigations.

ACKNOWLEDGEMENTS

The Know-Center is funded by the Austrian Competence Center program K plus under the auspices of the Austrian Ministry of Transport, Innovation and Technology (http://www.ffg.at/index.php?cid=95) and by the State of Styria.

REFERENCES

[1] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto, *Modern Information Retrieval*, Addison-Wesley Longman Publishing Co., Inc., 1999.

[2] S. Berretti, A. Del Bimbo, and P. Pala, 'A graph edit distance based on node merging', in *Image and Video Retrieval: Third International Conference, CIVR 2004*, volume 3115 of *LNCS*, pp. 464–472, Dublin, Ireland, (July 21-23 2004). Springer.

[3] Horst Bunke and Kim Shearer, 'A graph distance metric based on the maximal common subgraph', *Pattern Recognition Letters*, **19**(3-4), 255–259, (1998).

[4] Dieter Fensel, James A. Hendler, and Henry Lieberman, *Spinning the Semantic Web Bringing the World Wide Web to Its Full Potential*, MIT Press, 2005.

[5] Manuel J. Fonseca, B. Barroso, and Joaquim A. Jorge, 'Retrieving clipart images by content', in *Image and Video Retrieval: Third International Conference, CIVR 2004*, volume 3115 of *LNCS*, pp. 500–507, Dublin, Ireland, (July 21-23 2004). Springer.

[6] Michael R. Garey and David S. Johnson, *Computers and Intractability*, W.H. Freeman and Company, New York, 1979.

[7] Harald Kosch, *Distributed Multimedia Database Technologies*, CRC Press, Nov. 2003.

[8] Sven Meyer zu Eissen, Benno Stein, and Martin Potthast, 'The suffix tree document model revisited', in *Proceedings of the I-Know '05 5th International Conference on Knowledge Management*, pp. 596–603, Graz, Austria, (July 2005). J.UCS.

[9] S. E. Robertson and S. Walker, 'Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval', in *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 232–241, New York, NY, USA, (1994). Springer-Verlag New York, Inc.

[10] Stephen Robertson, Hugo Zaragoza, and Michael Taylor, 'Simple bm25 extension to multiple weighted fields', in *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pp. 42–49, New York, NY, USA, (2004). ACM Press.

[11] Dennis Shasha, Jason T. L. Wang, and Rosalba Giugno, 'Algorithmics and applications of tree and graph searching', in *PODS '02: Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pp. 39–52. ACM Press, (2002).

[12] Ali Shokoufandeh, Sven J. Dickinson, K. Siddiqi, and S.W. Zucker, 'Indexing using a spectral encoding of topological structure', in *Conference on Computer Vision and Pattern Recognition, IEEE Computer Society*, volume 2, pp. 491–497, USA, (June 1999).

[13] R. F. Simmons, 'Storage and retrieval of aspects of meaning in directed graph structures', *Commun. ACM*, **9**(3), 211–215, (1966).

[14] John F. Sowa, 'Semantics of conceptual graphs', in *Proceedings of the 17th annual meeting on Association for Computational Linguistics*, pp. 39–44, Morristown, NJ, USA, (1979). Association for Computational Linguistics.

[15] Gabriel Valiente, *Algorithms on Trees and Graphs*, Springer, Berlin, Germany, September 2002.

[16] Takashi Washio and Hiroshi Motoda, 'State of the art of graph-based data mining', *SIGKDD Explor. Newsl.*, **5**(1), 59–68, (2003).

[17] Xifeng Yan, Philip S. Yu, and Jiawei Han, 'Graph indexing: a frequent structure-based approach', in *SIGMOD '04: Pro-*

ceedings of the 2004 ACM SIGMOD international conference on Management of data, pp. 335–346. ACM Press, (2004).

[18] Jiwei Zhong, Haiping Zhu, Jianming Li, and Yong Yu, 'Conceptual graph matching for semantic search', in *ICCS '02: Proceedings of the 10th International Conference on Conceptual Structures*, pp. 92–196, London, UK, (2002). Springer-Verlag.