Spreading Activation: A Fast Calculation Method for Text Centroids

Mario M. Kubek
Chair of Communication Networks
University of Hagen
Universitätsstr. 27, Hagen, Germany
mario.kubek@fernuni-hagen.de

Thomas Böhme
Institut für Mathematik
Technische Universität Ilmenau
Weimarer Straße 25, Ilmenau, Germany
thomas.boehme@tu-ilmenau.de

Herwig Unger
Chair of Communication Networks
University of Hagen
Universitätsstr. 27, Hagen, Germany
herwig.unger@fernuni-hagen.de

ABSTRACT

Centroids are comfortable instruments to represent queries and whole texts by single descriptive terms. They can be used to determine the similarity of textual contents and to (hierarchically) cluster sets of documents. However, their computation strictly following the concept's definition may use a plenty of time and hinder any practical application. A more demonstrative view on the meaning and topological interpretation of the definition leads to the derivation of a graph-based algorithm using the well-known spreading activation technique, which is described in this contribution. The experimental results obtained using co-occurrence graphs of varying sizes underline the high performance of this method which is -last but not least- brought about by its clear local working principle.

CCS Concepts

•**Mathematics of computing** → **Graph algorithms;** •**Information systems** → **Query representation;**

Keywords

centroid term, co-occurrence graph, spreading activation, query diversity

1. MOTIVATION

Text centroids -inspired from the centre of mass in physics- and their application have been introduced in [1] and further articles like [2] have been used to deeply discuss their properties and to derive some interesting applications.

Differing from other methods, the determination of text representing centroids depends on some general knowledge and experience of a user, agent or program represented in the condensed structure of a co-occurrence graph and a defined metrics on it.

ICCIP'17, November 24–26, 2017, Tokyo, Japan.

ACM ISBN 978-1-4503-5365-6/17/11…$15.00

DOI: https://doi.org/10.1145/3162957.3163014

Any two words w_i and w_j are called co-occurrents, if they appear together in one sentence (or any other well-defined environment, context or window). This co-occurrence relation may be used to define a respective graph $G = (W, E)$. Therefore, the set words of a document corresponds to the set of nodes $w_a \in W$ and two nodes are connected by an edge $(w_a, w_b) \in E$, if w_a and w_b are co-occurrents. A weight function $g((w_a, w_b))$ can be introduced to represent the frequency of a co-occurrence in a document, while usually only co-occurrences of a high significance $\sigma > 1$, $\sigma \leq g((w_a, w_b))$ are taken into account.

Distances can be defined on G, if two words are considered to be closely related, if they appear often together, i.e. $g((w_a, w_b))$ is big enough. If $g(w_a, w_b) > 0$, the distance $d(w_a, w_b)$ of the co-occurring words w_a and w_b is defined by

$$d(w_a, w_b) = \frac{1}{g((w_a, w_b))}.$$

For word pairs that do not co-occur, the shortest path $p = \{(w_a, w_2), (w_2, w_3), .., (w_k, w_b)\}$ with $(w_i, w_{i+1}) \in E$ is considered and the distance is defined by

$$d(w_a, w_b) = \sum_{i=1}^{k} d((w_i, w_{i+1})).$$

Consequently, the distance of two words w_a and w_b being isolated nodes or situated in two, not connected sub-graphs is set to

$$d(w_a, w_b) = \infty.$$

By replacing frequencies of co-occurrences with distances, the co-occurrence graph is transformed into an isomorphic (word) distance graph.

In order to determine a centroid term $\chi(D)$ of a document D, the set of N words $W(D) = w_1, w_2, .., w_N \in D$ (filled by starting with the most frequent words) is considered with the nodes in a fully connected (sub-)component of the co-occurrence graph, such that pairwise distances $d(w_a, w_b) < \infty$ are guaranteed.

The centroid term $\chi(D)$ of a document is the term with the minimal average distance to all words of the document represented in $W(D)$[1], i.e. $d(D, \chi(D)) = MINIMAL$ for

$$d(D, t) = \frac{\sum_{i=1}^{N} d(w_i, t)}{N}.$$

[1]Note, that not necessarily $\chi \in W(D)$.

The direct application of this definition within a calculation algorithm would require to check all nodes of the fully connected (sub-)component of the co-occurrence graph, whether they fulfil the above defined minimum property. This results in an average complexity of $\mathbf{O}(|W|^3)$. Since co-occurrence graphs may contain up to 500,000 nodes (including nouns, names, composites etc.), significant calculation times in the range of several minutes may appear even on powerful machines.

2. CONCEPTUAL APPROACH

2.1 Idea

To understand the idea of the herein presented method which utilises the spreading activation technique [3] to address this problem, the physical correspondence of the text centroids, i.e. the centre of mass must be considered again, as presented in [1]. In a physical body, the centre of mass is usually expected to be inside a convex hull line of the (convex or concave) body, in case of a homogeneous one and is situated more or less in its middle (Figure 1[2]).

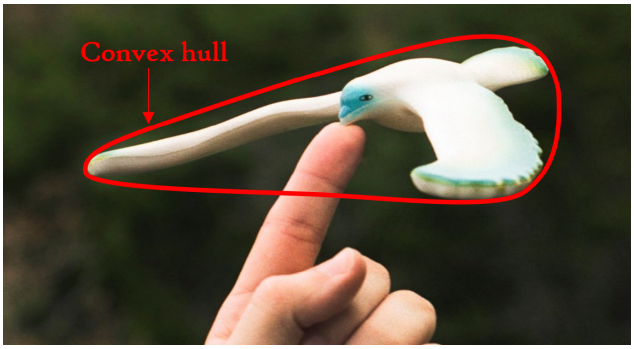

Figure 1: A convex hull curve of a bird-toy and its centre of mass

If a set of uniform, discrete mass points in a 2-dimensional, Euclidean plane with an underlying rectangular grid is considered, one would try to fix the centre or mass in the intersection of concentric cycles of the same radius around those points (Figure 2).

Things may look more complex in a usual co-occurrence graph, since it can normally not be embedded in a 2- or 3-dimensional space, which humans can easily imagine. However, similar ideas of a neighbourhood allocation have already been used to provide a graph clustering method [4]. The so-called *Chinese Whispers* algorithm [5] is another interesting and related solution for efficient graph clustering that relies on a label propagation technique. The algorithm in the next subsection adapts the idea described above and can be applied on large co-occurrence graphs quite fast.

2.2 Algorithm

Usually, the co-occurrence graph can be kept on every machine. Therefore, the calculation of text centroids can be carried out in a local, serial manner. If only shortest paths

[2]Modified from https://commons.wikimedia.org/wiki/File:Bird_toy_showing_center_of_gravity.jpg, original author: APN MJM, Creative Commons licence: CC BY-SA 3.0

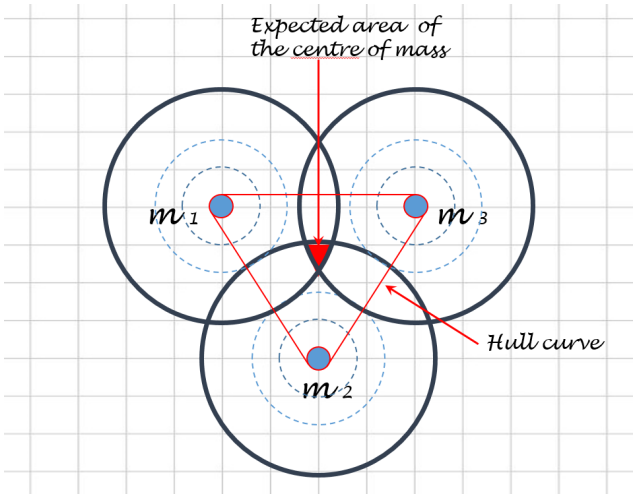

Figure 2: Locating the centre of mass in the 2-dimensional plane

are considered between any two nodes in the co-occurrence graph, a metric system is built.

In the following considerations, a query set Q of s words $Q = \{w_1, w_2, .., w_s\}$ shall be considered. Q is called a **query set**, if it contains (usually after a respective preprocessing) only words $w_1, w_2, .., w_s$, which are nodes within a single, connected component of the co-occurrence graph $G = (W, E)$ denoted by $G' = (W', E')$.

A fast calculation method for the centroid term $\chi(Q)$ of a query set Q shall be presented at this point.

For computation purposes, a vector $\bar{v}(w') = [v_1, v_2, .., v_s]$ is assigned to each $w' \in W'$ with the components being initialised to 0. With this preparations, the following, **spreading activation algorithm** is executed.

1. Determine (or estimate) the maximum of the shortest distances d_{max} between any pair (w_i, w_j) with $w_i, w_j \in Q$, i.e. let
$$d_{max} = sup(d(w_i, w_j)|_{(w_i,w_j)\in Q\times Q})$$

2. Choose a radius $r = \frac{d_{max}}{2} + \Delta$, where Δ is a small constant of about $0.1 \cdot d_{max}$ ensuring that an overlapping area will exist.

3. Apply (or continue) a breadth-first-search algorithm from every $w_i \in Q$ and activate (i.e. label) each reached, recent node w' for every w_i by
$$\bar{v}(w')[v_i] = d(w_i, w') \leftrightarrow d(w_i, w') \leq r.$$
Stop the activation, if no more neighbourhood nodes with $d(w_i, w') \leq r$ can be found.

4. Consider all nodes $w' \in W'$ with
$$\forall i, \quad i = 1..s : \bar{v}(w')[v_i] \neq 0$$
and choose among them the node with the **minimal**
$$\sum_{i=1}^{s} \bar{v}(w')[v_i]$$
to be the centroid $\chi(\{w_1, w_2, .., w_s)\}$.

5. If no centroids found, set $r := r + \Delta$ and GoTo 3, otherwise *STOP*.

The greatest benefit of the described method is that it generally avoids the 'visit' of all nodes of the co-occurrence graph as it solely affects the local areas around the query terms $w_1, w_2, .., w_s \in Q$.

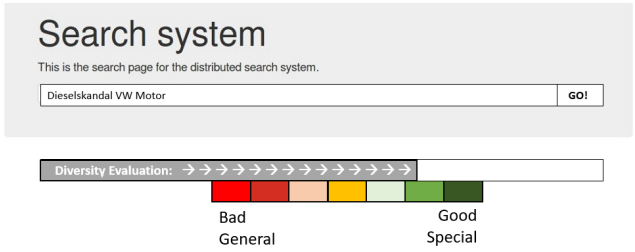

Figure 3: Using the query diversity as a user's guide

The supremum $sup(d(w_i, w_j)|_{\forall(w_i,w_j)\in(Q\times Q)})$ of a search query Q is called the **diversity** of a (search) query. The smaller the diversity is, the more a query targets a designated, narrow topic area, while high values of the diversity mark a more general, common request. This may be used to provide additional guidance and support for users during interactive search sessions on the usually keyword-oriented search engines (see Figure 3).

3. EXPERIMENTAL EVALUATION

In this section, the performance of the presented algorithm will be evaluated in a number of experiments. All measurements have been performed on a Lenovo Thinkpad business-class laptop equipped with an Intel Core i5-6200U CPU and 8 GB of RAM to show that the algorithm can even be successfully applied on non-server hardware. The four datasets[3] used to construct the co-occurrence graphs consist of either 100, 200, 500 or 1000 topically classified (topical tags assigned by their authors) online news articles from the German newspaper "Süddeutsche Zeitung". In order to build the (undirected) co-occurrence graphs, linguistic preprocessing has been applied on these documents whereby sentences have been extracted, stop words have been removed and only nouns (in their base form), proper nouns and names have been considered. Based on these preparatory works, co-occurrences on sentence level have been extracted. Their significance values have been determined using the Dice coefficient [6]. These values and the extracted terms are persistently saved in an embedded Neo4j (https://neo4j.com) graph database using its property-value store provided for all nodes (represent the terms) and relationships (represent the co-occurrences and their significances).

3.1 Exp. 1: Average Processing Time

In the first sets of experiments presented here, the goal is to show that for automatically generated queries of five different sizes (queries consisting of two to six terms) in six different ranges of diversity the average processing time to find their centroid terms is low when the spreading activation method is applied. The queries and the used co-occurrence graph have been generated using the dataset "Corpus-100" from which 4331 terms have been extracted. In order to determine the average processing time for each query size and for the six ranges of diversity, 20 different queries have been generated for these two parameters. Therefore, altogether 600 queries have been created.

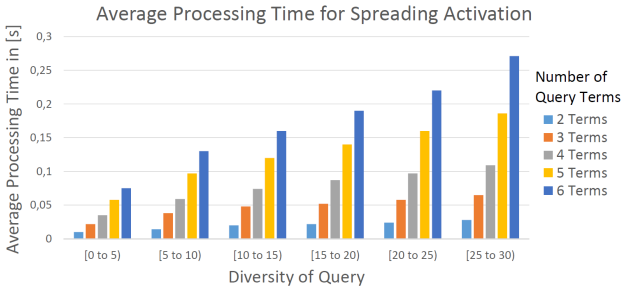

Figure 4: Average processing time for spreading activation

Figure 4 shows that even for an increased number of query terms and diversity values the average processing time stays low and increases only slightly. As the average processing time stays under half a second for all cases, the algorithm is clearly suited for application in interactive search systems.

In order to demonstrate the great improvement in processing time of this algorithm, the original algorithm (as the direct application of the centroid definition given above) has been run on five queries consisting of two to six terms in the diversity range of [10-15) as a comparison while using the dataset "Corpus-100" to construct the co-occurrence graph as well. Table 1 presents the absolute processing times in [s]

Table 1: Processing times of the original algorithm

Number of query terms	2	3	4	5	6
Processing time in [s]	185	252	317	405	626

needed by the original algorithm to determine the centroid terms for those queries. Due to these high and unacceptable values, the original algorithm -in contrast to the herein presented one- cannot be applied in interactive search systems that must stay responsive at all times.

3.2 Exp. 2: Node Activation

The second set of experiments examines the average number of nodes activated when the presented algorithm is run starting from a particular centroid term of a query while restricting the maximum distance (this value is not included) from this starting node. As an example, for a maximum distance of 5, all activated nodes by the algorithm must have a smaller distance than 5 from the centroid. In order to be able to determine the average number of activated nodes, the algorithm has been started for 10 centroid terms under this restriction. The maximum distance has been varied (increased) from 1 to 25. For this experiment, the dataset "Corpus-100" has been used again.

As Figure 5 shows, the average number of activated nodes visibly and constantly rises starting from the maximum distance of 7 (40 activated nodes). At the maximum distance of 25, in average, 3780 nodes have been activated. The result also shows that for queries with a low diversity (and a therefore likely high topical homogeneity) the number of

[3]Interested readers may download these datasets (4.1 MB) from: http://www.docanalyser.de/sa-corpora.zip

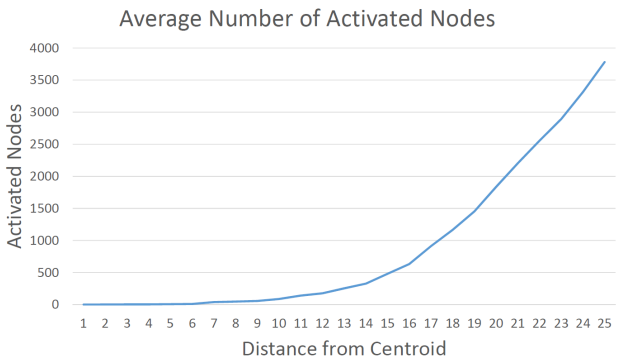

Figure 5: Average number of activated nodes

activated nodes will stay low as well. In this example, for a low maximum distance of 10, the average number of activated nodes is only 88 (2 percent of all nodes in the used co-occurrence graph). Therefore and as wished-for, the activation stays local, especially for low diversity queries.

3.3 Exp. 3: Growing Co-occurrence Graph

As document collections usually grow, the last sets of experiments investigate the influence of a growing co-occurrence graph on the processing time of the introduced algorithm. For this purpose, the datasets "Corpus-100", "Corpus-200", "Corpus-500" and "Corpus-1000" respectively consisting of 100, 200, 500 and 1000 news articles have been used to construct co-occurrence graphs of increasing sizes with 4331, 8481, 18022 and 30048 terms/nodes. Also, as the document collection should be growing, it is noteworthy to point out that corpora of smaller sizes are included in the corpora of larger sizes. For instance, the articles in dataset "Corpus-200" are included in both "Corpus-500" and "Corpus-1000", too. In order to conduct the experiments, four queries have been chosen: one query with two terms and a low diversity in the range of [5-10), one query with two terms and a high diversity in the range of [20-25), one query with six terms and a low diversity in the range of [5-10) and one query with six terms and a high diversity in the range of [20-25). For each of these queries and each corpus, the absolute processing time in [s] (in contrast to the previous experiments that applied averaging) to determine the respective centroid term has been measured.

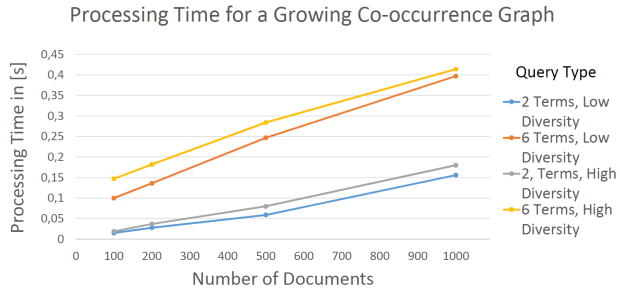

Figure 6: Influence of a growing co-occurrence graph

The curves in Figure 6 show an almost linear rise in processing time for all four queries and the four growing co-occurrence graphs. Besides the size of the co-occurrence graph used, the query size is of major influence on the processing time. While the query's diversity plays a rather secondary role at this, it can clearly be seen that -even initially- the processing times for the queries with a high diversity are higher than for their equal-sized counterparts with a low diversity. However, even in these experiments and especially for the query with six terms and high diversity and the largest co-occurrence graph of 30048 nodes, the processing time stayed low with 0.41 seconds. While these experiments showed that the processing time will understandably increase when the underlying co-occurrence graph is growing, its rise is still acceptable, especially when it comes to handle queries in a (graph-based) search system. The reason for this is again the algorithm's strict local working principle. Node activation will occur around the requested query terms only while leaving most of the nodes in the graph inactivated.

4. CONCLUSION

A new graph-based algorithm to determine centroid terms of queries and text documents in a fast manner has been presented. In three sets of experiments conducted on modern laptop hardware, its performance has been positively evaluated for application in search systems. Due to its local working principle, it can be efficiently applied even when no server hardware is used. As the integrated spreading activation technique can be independently executed for every single initially activated node/term (e.g. from a query), the algorithm's core steps can be performed in parallel, e.g. in separate threads. This makes an effective utilisation of potentially available multiple CPU cores possible. Future optimisations of this algorithm will therefore focus on its parallelisation.

5. REFERENCES

[1] M. M. Kubek and H. Unger. Centroid terms as text representatives. In *Proceedings of the 2016 ACM Symposium on Document Engineering*, DocEng '16, pages 99–102, New York, NY, USA, ACM, 2016.

[2] M. M. Kubek and H. Unger. Towards a librarian of the web. In *Proceedings of the 2Nd International Conference on Communication and Information Processing*, ICCIP '16, pages 70–78, New York, NY, USA, ACM, 2016.

[3] S. E. Preece. *A Spreading Activation Network Model for Information Retrieval*. PhD thesis, Champaign, IL, USA, 1981.

[4] H. Unger and M. Wulff. Cluster-building in p2p-community networks. In *International Conference on Parallel and Distributed Computing Systems, PDCS 2002, November 4-6, 2002, Cambridge, USA*, pages 680–685, 2002.

[5] C. Biemann. Chinese whispers: An efficient graph clustering algorithm and its application to natural language processing problems. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, TextGraphs-1, pages 73–80, Stroudsburg, PA, USA, Association for Computational Linguistics, 2006.

[6] L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, July 1945.