RESEARCH ARTICLE OPEN ACCESS

Improving Information Retrieval On The Web Using Clustering Approaches

Dr. A.Chandrabose1, T. Manivannan2, M. Jayakandan3, P. Ananthi4
[1]*Associate Professor* [2]*Assistant Professor* [3]*Assistant Professor* *Assistant Professor*

EDAYATHANGUDY G.S.PILLAY ARTS & SCIENCE COLLEGE-NAGAPATTINAM-611 002

Coresspondin author Dr. A.Chandrabose

ABSTRACT

As large amounts of digital information become more and more accessible. The ability to effectively find relevant information is increasingly important. Search engines have historically performed well at finding relevant information by relying primarily onlexical and word based measures. Similarly, standard approaches to organizing and categorizing large amounts of textual information have previously relied on lexical and word based measures to perform grouping or classification tasks. Quite often, however, these processes take place without respect to semantics, or word meanings. This is perhaps due to the fact that the idea of meaningful similarity is naturally qualitative, and thus difficult to incorporate into quantitative processes.

Keywords: our semantic distance metric can be used to improve document clustering in distance-based clustering algorithms queries

Date Of Submission:02-10-2018 Date Of Acceptance: 13 -10-2018

I. INTRODUCTION

In the last twenty years, digital information has become widely accessible on an unprecedented scale. As a result, several related, important problems have become the focus of many computer scientists and researchers. Some of these problems include how to best retrieve information that is relevant to a user's interest, as well as how to automatically organize large amounts of digital information. Many different approaches have been taken to address these problems, and some with a great deal of success. However, there remains significant room for improvement.

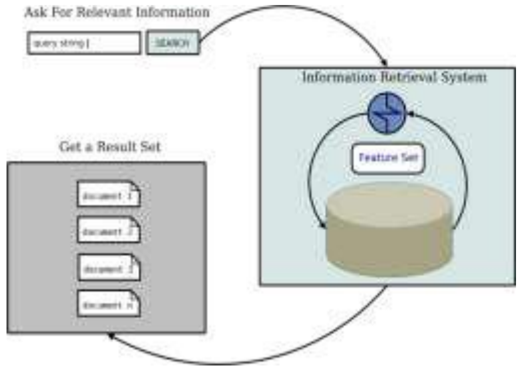


Figure 1.1Web search engines, such as Google,

A popular method of supplying user input to an information retrieval system is through keyword queries .Web search engines, such as Google, perform this task very well. Despite the success of existing information retrieval systems, the relationships between queries and documents have traditionally been limited to lexical analysis. This is evident in the very popular Vector Space Model (VSM) [53]. Essentially the vector space model represents documents as vectors, whose element values represent some lexical measure related to words (i.e., terms) in a document, such as term frequency and inverse document frequency (TF-IDF), which is derived by observing frequencies of terms in and across documents [52]. A VSM document retrieval system can be issued a keyword query, where documents are returned whose vector representation has the smallest distance from the vector representation of the query.

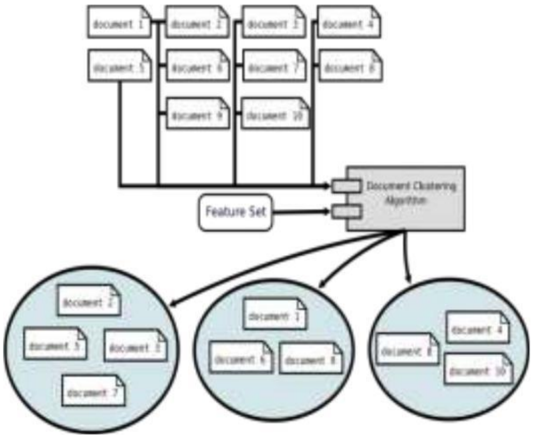


Figure 1.2 Structuring Semantic Relationships

II. PROCESS MECHANISM AND FOUNDATION

2.Ontologies

The specific structure that we use to represent hierarchical semantic relationships is an ontology. An ontology looks much like a tree structure, and expresses a taxonomy. While we will provide a method for quantitatively representing any type of hierarchical semantic relationships, our primary focus is devoted to relationships of hypernymy. Consider, for example, the words “calculator” and “computer” in the English language. Figure 3.1 shows a practical hypernymy tree for “calculator”, where each concept is a subclass of the concept that precedes it in the hierarchy.

2.1WordNet

For our experiments, we use version 3.0 of WordNet as the hypernymy ontology. WordNet version 3.0 contains 155,287 unique English words. Other such structures could be used with the methods we develop, as long as they describe hierarchical semantic relations, however we focus our efforts and experiments on usingsemantic relations, however we focus our efforts and experiments on using WordNet because of is rich and extensive semantic relationship hierarchy.

=>calculator
=>abacus
=>adder
=> adding machine, tantalizer, totaliser
=>counter, tabulator =>pulse counter

=>scaler
=>hand calculator, pocket calculator
=> Napier’s bones, Napier’s rods
=>quipu
=>subtracter

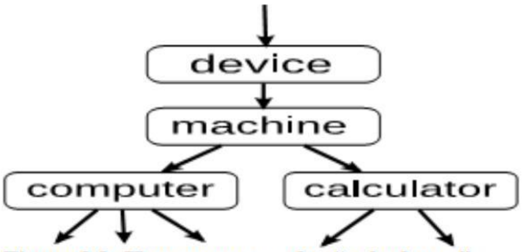


Figure 2.1: Hyponymy tree for “calculator.”

III. MATHEMATICAL FOUNDATION

The contributions of this thesis depend on a method for computing word-to-word semantic distances. Our work is based on the distance proposed in [22]. We give a short summary here of its design. The lexicon is mapped into a complete metric space where there exists a natural inner product, which is then used as the distance metric. This topological mapping allows for a distance metric that quantitatively represents the qualitative relationships specified by the lexicon.

3.1Dual Spaces

The mappings of these chains are continuous under the order topology, so that “close” concepts in the graph are mapped to “close” real numbers, as one would expect. We then exploit the duality that exists between the space of conceptual elements (i.e., the WordNet nouns) and the space of chain-functions, as depicted in Figure 3.4. Finally, an inner product is adopted on the function space that is consistent with the original order topology, and use the inner product on the conjugate representations of the concept-elements to define a metric as the direction cosine distance in the dual space.

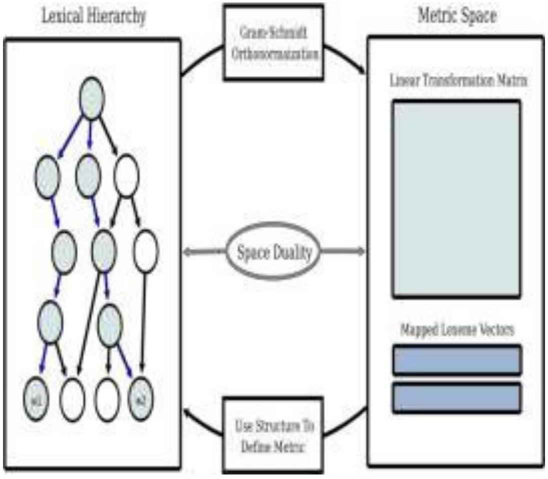

Lexical Hierarchy
Gram-Schmidt Orthonormalization
Metric Space
Linear Transformation Matrix
Space Duality
Mapped Lexeme Vectors
w1
w2
Use Structure To Define Metric

Figure 3.1: Dual spaces for computing effective semantic distance given a lexicon.

3.2Building the Metric Space

Let c be some maximal chain and let q be the length of c. We define the function m_c:c → R by

$$m_c(c^i) = \frac{i-1}{q-1}$$

where$c^i$is the ith element of c, ordered from root to leaf. Next, define the function f_c : N → [0,1] by:

$f_c(n) = m_c(c^{in})$

where$c^{in}$is the point of intersection of n closest to the leaf (lowest), via some chaincontainingn, with c.

We illustrate the method with the simple directed acyclic graph G of Figure 3.5. For G, the set of all maximal chains comprises our set of basis chains, while for larger graphs it may be more practical to choose a spanning set of basis chains.

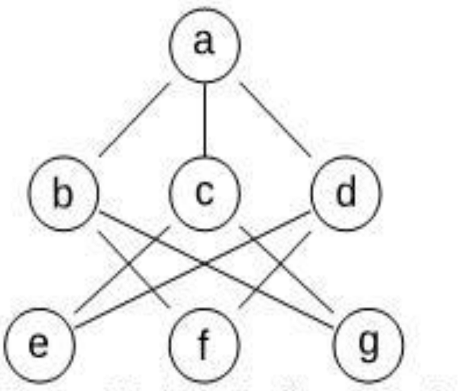


Figure 3.2: A simple directed graph.

For G, these chains are c_1 = {e,c,a}, c_2 = {e,d,a}, c_3 = {f,b,a}, c_4 = {f,d,a}, c_5 = {g,b,a}, and c_6 = {g,c,a}. The value of f_{c4}(e) is .5, since the earliest point of intersection of any chain containing e with c_4 is halfway down c_4. The complete action of the functions {f_{ck}} on G is shown in Figure 3.6

	fc_1	fc_2	fc_3	fc_4	fc_5	fc_6
a	0	0	0	0	0	0
b	0	0	.5	0	.5	0
c	.5	0	0	0	0	.5
d	0	.5	0	.5	0	0
e	1	1	0	.5	0	.5
f	0	.5	1	1	.5	0

The set of functions {f_{ck}} naturally spans a vector space consisting of all linear combinations of the f_{ck}'s. We now find an orthonormal set of functions that provides a basis for that vector space, and produce a linear transformation from N to the corresponding conjugates in the function space with respect to the basis (hence, aninner-product preserving representation of the original elements of G).

In the particular case of G, the Gram-Schmidt algorithm is employed to compute an orthonormal basis {w_i} from {f_{ck}}. The Gram-Schmidt process takes a finite, linearly independent set of vectors (i.e., a basis) and produces a new set of ortho normal vectors that span the same space.

Note that f_6 is in the span of the other functions, so that {f_1,f_2,f_3,f_4,f_5} is the linearly independent set of vectors to start from.

The Gram-Schmidt algorithm is as follows:

u1 = f1

$\Rightarrow w_1 = u_1/\|u_1\|$ $u_2 = f_2 - < f_2,w_1 > w_1 \Rightarrow w_2 = u_2/\|u_2\|$ u_3

$= f_3 - < f_3,w_2 > w_2 - < f_3,w_1 > w_1$

$\Rightarrow w_3 = u_3/\|u_3\|$ $u_4 = f_4 - < f_4,w_3 > w_3 - < f_4,w_2 > w_2 - <$

$f_4,w_1 > w_1$

$\Rightarrow w_4 = u_4/\|u_4\|$ $u_5 = f_5 - < f_5,w_4 > w_4 - < f_5,w_3 > w_3 - <$

$f_5,w_2 > w_2 - < f_5,w_1 > w_1$

$\Rightarrow w_5 = u_5/\|u_5\|$

This yields the following matrix form:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{||w_5||} \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -<f_5,w_1> & -<f_5,w_2> & -<f_5,w_3> & -<f_5,w_4> & 1 \end{bmatrix} \times$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{||w_4||} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -<f_4,w_1> & -<f_4,w_2> & -<f_4,w_3> & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{||w_3||} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -<f_3,w_1> & -<f_3,w_2> & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{||w_2||} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -<f_2,w_1> & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times$$

$$\begin{bmatrix} \frac{1}{||w_1||} & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{bmatrix}$$

The various matrices are referred to as E_1,E_2... E_{10}, labeled from right to left. The steps deriving $w_i$correspond to the elementary operations E_{2i}, E_{2i-1}. We thus obtain an orthonormal basis {w_i} via the matrix Q = Qi E_i. This gives:

$$Q^{-1} = \begin{bmatrix} 1.2247 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.8165 & 0.9129 & 0.0000 & 0.0000 & 0.0000 \\ 0.2041 & 0.3651 & 1.1511 & 0.0000 & 0.0000 \\ 0.4028 & 1.0042 & 0.4778 & 0.3110 & 0.0000 \\ 0.4082 & -0.0913 & 1.0425 & -0.2351 & 0.4277 \end{bmatrix}$$

3.3Word-to-Word Semantic Distance

The method described allows us to compute effective, quantitative, word-level semantic distance. Figure 3.7 outlines the algorithm, which we use as the foundation for our document-level semantic distance metric.

1. Select a set of basis chains B = {c_k}
2. Orthonormalize B and derive the matrix Q = Qi E_i, where the E_i's are the elementary row operations of the Gram-Schmidt procedure
3. Given a new lexeme, l

(a) Derive the vector ~l using the function $m_{ck}$for each basis chain c_k(i.e., the coordinates of a chain containing l in B)

(b) Compute the transformed vector $\sim l^{'} = \sim lQ^{-1}$

4. For any two lexemes, l_1 and l_2

(a) Compute$\vec{l'_1}$ and $\vec{l'_2}$ as per step 3

$$(b)\ d(l_1, l_2) = arccos\frac{<\vec{l'_1},\vec{l'_2}>}{||\vec{l'_1}||||\vec{l'_2}||}$$

Figure 3.3: Word-to-word semantic distance computation by Jensen et al. Word-to-Word Preliminary Resu

w_1	w_2	humans	d(w_1,w_2)	w_1	w_2	humans	d(w_1,w_2)
cord	smile	0.02	**33.16**	car	journey	1.55	**27.65**
rooster	voyage	0.04	**35.92**	cemetery	mound	1.69	**7.2**
noon	string	0.04	**29.84**	glass	jewel	1.78	**1.65**
fruit	furnace	0.05	**5.56**	magician	oracle	1.82	**5.96**
autograph	shore	0.06	**27.01**	crane	implement	2.37	**0.84**
automobile	wizard	0.11	**25.61**	brother	lad	2.41	**3.74**
mound	stove	0.14	**24.37**	sage	wizard	2.46	**13.38**
grin	implement	0.18	**32.13**	oracle	sage	2.61	**8.82**
asylum	fruit	0.19	**24.69**	bird	crane	2.63	**0.00**
asylum	monk	0.39	**33.05**	bird	cock	2.63	**0.00**
graveyard	madhouse	0.42	**7.56**	food	fruit	2.69	**1.13**
glass	magician	0.44	**24.8**	brother	monk	2.74	**0.00**
boy	rooster	0.44	**27.43**	asylum	madhouse	3.04	**0.21**
cushion	jewel	0.45	**24.41**	furnace	stove	3.11	**1.7**
monk	slave	0.57	**20.57**	magician	wizard	3.21	**0.00**
asylum	cemetery	0.79	**7.48**	hill	mound	3.29	**0.00**
coast	forest	0.85	**17.44**	cord	string	3.41	**0.00**

grin	lad	0.88	**29.32**	glass	tumbler	3.45	**1.21E-6**
shore	woodland	0.90	**12.64**	grin	smile	3.46	**0.00**
monk	oracle	0.91	**19.01**	serf	slave	3.46	**4.02**
boy	sage	0.96	**18.67**	journey	voyage	3.58	**0.00**
automobile	cushion	0.97	**4.18**	autograph	signature	3.59	**1.21E-6**
mound	shore	0.97	**10.34**	coast	shore	3.60	**2.86**
lad	wizard	0.99	**6.85**	forest	woodland	3.65	**1.21E-6**
forest	graveyard	1.00	**12.13**	implement	tool	3.66	**0.4**
food	rooster	1.09	**30.65**	cock	rooster	3.68	**8.54E-7**
cemetery	woodland	1.18	**12.13**	boy	lad	3.82	**0.00**
shore	voyage	1.22	**27.11**	cushion	pillow	3.84	**0.2**
bird	woodland	1.24	**13.34**	cemetery	graveyard	3.88	**8.54E-7**
coast	hill	1.26	**8.07**	automobile	car	3.92	**0.00**
furnace	implement	1.37	**1.72**	midday	noon	3.94	**0.00**
crane	rooster	1.41	**4.72**	gem	jewel	3.94	**0.00**
hill	woodland	1.48	**12.51**				

Table 3.1: 65 noun pairs provided by Rubenstein and Goodenough along with human similarity scores, and our distance measures [49]. The correlation of our scores with the humans was **-0.802**.

Measure	**Correlation w/ humans**
Leacock and Chodorow	.838
Lin	.819
Jensen et al. [22]	**.802**
Hirst and St-Onge	.786
Jiang and Conrath	.781
Resnik	.779

Table 3.2: Our correlation performance in comparison to five other approaches.

3.4 Hausdorff Semantic Distance for Documents

To maintain consistency with our basic approach at the word-level, driven by topo-

Logicalarguments, we adopt the Hausdorff distance, defined as follows.

$H(d_1,d_2) = \max_{n \in d1}(\min_{n' \in d2}(dist(n,n')))$

$H(d_2,d_1) = \max_{n' \in d2}(\min_{n \in d1}(dist(n,n')))$

$Hausdorff(d_1,d_2) = \max\{H(d_1,d_2),H(d_2,d_1)\}$

For each noun in a document, the minimum word-level distance to all nouns in the other document is found. Then, the maximum of all of these minimum distances is taken as the distance from the first document to the second one, as depicted in Figure 3.3. This process is repeated reversing the order of the documents, since that measure is clearly not symmetric. Finally, the maximum of the two distances is the Hausdorff distance between the documents.

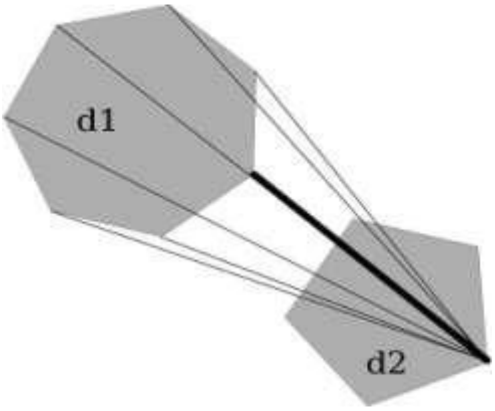


Figure 3.4: Visualizing a Hausdorff distance from document d_1 to d_2.

```
defhausdorff_dist_aux(doc_a, doc_b):
max_ab = -Inf for t_x in doc_a: min_ab = Inf
fort_y in doc_b: dist = d(t_x, t_y) if dist<min_ab:
min_ab = dist
ifmin_ab>max_ab:
max_ab = min_ab return max_ab
defhausdorff_dist(doc_a, doc_b):
return max(hausdorff_dist_aux(doc_a, doc_b),
hausdorff_dist_aux(doc_b, doc_a))
```

3.5Document Distance Through Aligned Word Clusters

Typical approaches to determining a suitable distance measure between documents tend to make the implicit assumption that documents have a single semantic topic. What if that assumption is violated? Consider the following contrived example of two documents, each comprised of two semantic topics, described by word clusters (i.e., sets of words that have a small word-level semantic distance from each other): d_1 =

{ [“house”, “apartment”], [“book”] } and d_2 = { [“condo”, “mansion”], [“basketball”] }. Using the Hausdorff distance, we might find that the distance between these two documents is measured by the distance between “book” and “mansion” (i.e, their word-to-word distance is the maximum of all the minimum word-to-word distances for the two documents). Then, the documents would appear rather distant from each other. On the other hand, if we somehow align the word clusters and compute the distance between them, we would find that the distance between the first clusters in d_1 and d $_2$ is rather small, and hence, the documents could be deemed rather close. In this section, we describe a method to handle the clustering of documents withmultiple semantic topics.

IV. IMPLEMENTATION AND FLOW CONTROL

Word Cluster Alignment

One advantage of this method is that it runs slightly faster than the normal symmetric Hausdorff distance. While it is bounded by the same O(n × m) (where n and m represent the number of words in d_1, d_2 respectively), the preceding multiplicative constants are smaller because the number of cluster comparisons decreases as clusters are aligned and become unavailable. Another advantage of this approach is that it attempts to handle documents with multiple, and potentially differing, semantic topics. This approach, however, is not as theoretically satisfying as the Hausdorff distance, in the context of a topological mapping (the basis for our word-level semantic distance metric). A challenge it presents is determining the best number of word clusters in documents.

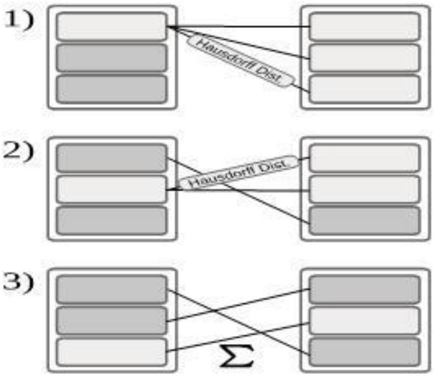
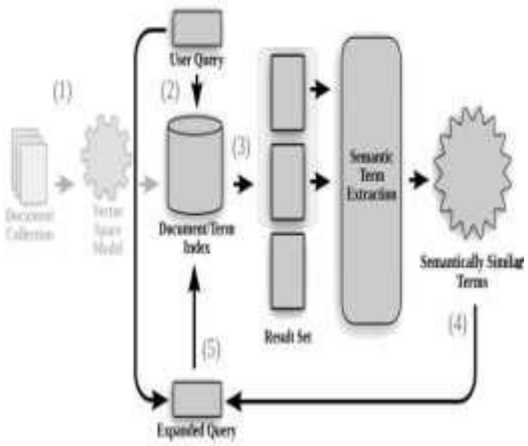
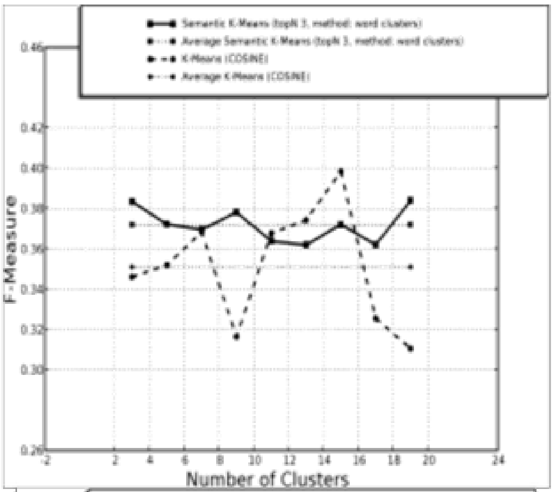


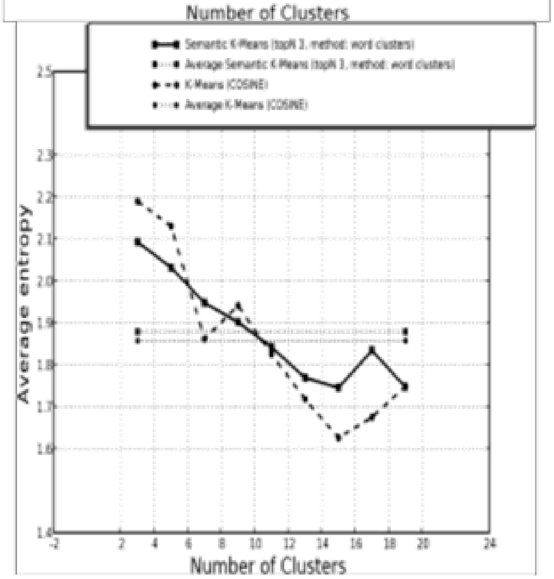
Figure 3.5: Aligning word clusters. Once all clusters have been aligned, the distances are summed to form a document level distance metric.

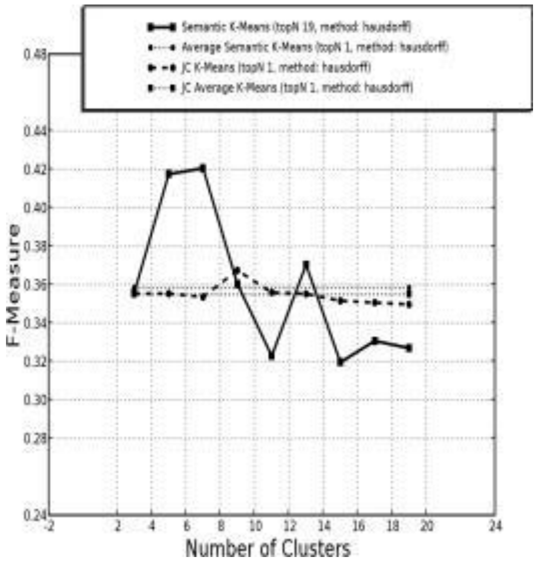


V. CONCLUSION AND FUTURE WORK

We have shown that using our document-level distance metric to re-rank query results yielded higher levels of precision for the 50 queries of the very large, and real world Aquaintdata set. We have also shown that the use of our semantic distance metric in expanding user keyword queries also improves precision and recall.







The most significant contribution of this thesis is our design, implementation, and use of a novel, effective, and theoretically sound document-level semantic distance metric. This distance metric makes possible document-level quantitative analysis for normally qualitative word semantics. We built our document-level distance metric from our implementation of an effective word-level distance metric, and successfully showed how it could be used to improve results for real world problems related to document retrieval and clustering.

Additionally, we define a secondary method for expanding user keyword queries using Latent Dirichlet Allocation (LDA). While this method differs in its approach from our knowledge-based, semantic distance approach, it is nonetheless a significant contribution as it was found to be both novel (to the best of our knowledge) and highly effective in improving precision and recall. Interesting research related to our semantic distance metric remains for future work. Specific items of future work include:Utilizing semantic distance for text classification.

- Applying similar experiments to those described in this thesis, to data sets where hypernymy/hyponymy are more evident, such as query-by-example data sets. One could imagine a keyword query system that allowed users to search for images by providing keywords that describe the concepts in the image. A search for “buildings”, or similar example type concepts, may be a setting where our hypernymy/hyponymy WordNet semantic distance would be very well suited.
- Exploring alternate knowledge structures, such as the Wikipedia category hierarchy, as an alternative to WordNet.
- Forming clusters, using our distance metric, for the purpose of initializing probability distributions for words and clusters that can be used to seed the Expectation Maximization clustering method.

REFERENCES

[1]. Thelongmandefiningvocabulary,March2007. http://home.earthlink.net/ neilbawd/longman.txt.

[2]. M. Baziz, M. Boughanem, and N. Aussenac-Gilles. A conceptual indexing approach for the trec robust task. In The Fourteenth Text REtrieval Conference (TREC 2005) Proceedings, 2005.

[3]. D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. In Journal of machine Learning Research 3, 2003.

[4]. D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation. Journal of Machine Learning Research, 3:993–1022, 2003.

[5]. F. Blei, M. Ester, and X. Xu. Frequent term-based text clustering. In Proceedings of the 8th International Conference on Knowledge Discovery and Data Mining, pages 436–442, 2002.

[6]. T. Bogers and A. van den Bosch. Authoritative re-ranking of search results.

[7]. Advances in Information Retrieval, 3936/2006:519–522, 2006.

[8]. A. Budanitsky and G. Hirst. Evaluating wordnet-based measures of lexical semantic distance. Computational Linguistics, 32(1):13–47, 2006.

[9]. S. Bttcher, C.L.A. Clarke, and P.C.K.Yeung. Index pruning and result reranking: Effects on ad-hoc retrieval and named page finding. In The Fifteenth Text REtrieval Conference (TREC 2006) Notebook, page 237, 2006.

[10]. B. Choudhary and P. Bhattacharyya. Text clustering using semantics. In Proceedings of the 11th International World Wide Web Conference, 2002.

[11]. D.R. Cutting, D.R. Karger, J.O. Pedersen, and J.W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In Proceedings of the 15th International Conference on Research and Development in Information Retrieval, pages 318–329, 1992.

[12]. Hal Daum´e III and Daniel Marcu. Bayesian query-focused summarization. In Proceedings of the Conference of the Association for Computational Linguistics (ACL), Sydney, Australia, 2006.

[13]. S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. Journal of the

American Society for Information Science, 41(6), 1990.

[14]. B. Everitt. Cluster Analysis. John Wiley & Sons, Inc., 1993.

[15]. D. Fisher. Knowledge acquisition via incremental conceptual clustering. Machine Learning, 2:139–172, 1987.

[16]. B. Fung, K. Wang, and M. Ester. Hierarchical document clustering using frequent itemsets. In Proceedings of the SIAM International Conference on Data Mining, pages 59–70, 2003.

[17]. J.H. Gennari, P. Langley, and D. Fisher. Models of incremental concept formation. Artificial Intelligence, 40:11–61, 1989.

[18]. P.V. Henstock, D.J. Pack, Y.-S. Lee, and C.J. Weinstein. Toward an improved concept-based information retrieval system. In Proceedings of the 24th Annual International Conference on Research and Development in Information Retrieval, pages 384–385, 2001.

[19]. T. Hofmann. Probabilistic latent semantic indexing. In Proceedings of the 22nd International Conference on Research and Development in Information Retrieval, pages 50–54, 1999.

[20]. A. Hotho, S. Staab, and A. Maedche. Ontology-based text clustering. In Proceedings of the IJCAI Workshop on Text Learning: Beyond Supervision, 2001.

[21]. L. Hubert and P. Arabie. Comparing partitions. Journal of Classification, 2(1):193–218, 1985.

[22]. A.K. Jain and R.C. Dubes. Algorithms for Clustering Data. Prentice-Hall, Inc., 1988.

[23]. D. Jensen, C. Giraud-Carrier, and N. Davis. A method for computing lexical semantic distance using linear functionals. In Journal of Web Semantics, 2007.

[24]. DOI:http://dx.doi.org/10.1016/j.websem.2007.11.001.

[25]. Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy, 1997.

[26]. S.C. Johnson. Hierarchical clustering schemes. Psychometrika, 2:241–254, 1967.

[27]. L. Kaufman and P.J. Rousseeuw. Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley & Sons, Inc., 1990.

[28]. S.-B. Kim, H.-C. Seo, and H.-C. Rim. Information retrieval using word senses: Root sense tagging approach. In Proceedings of the 27th Annual International Conference on Research and Development in Information Retrieval, pages 258– 265, 2004.

[29]. J. Kogan, M. Teboulle, and C. Nicholas. The entropic geometric means algorithm: An approach to building small clusters for large text datasets. In Proceedings of the ICDM Workshop on Clustering Large Data Sets, pages 63–71, 2003.

[30]. T. Kohonen. Self-organized formation of topologically correct feature maps.

[31]. Biological Cybernetics, 43(1):59–69, 1982.

[32]. T. K. Landauer, P. W. Foltz, and D. Laham. An introduction to latent semantic analysis. Discourse Processes, 25:259–284, 1998.

[33]. K. Lang. News weeder: Learning to filter netnews. In Proceedings of the 12th International Conference of Machine Learning, pages 331–339, 1995.

[34]. B. Larsen and C. Aone. Fast and effective text mining using linear-time document clustering. In Proceedings of the 5th International Conference on Knowledge Discovery and Data Mining, pages 16–22, 1999.

[35]. L. Lebart and M. Rajman. Handbook of Natural Language Processing Computing Similarity. Marcel Dekker, Inc., 2000.

[36]. A. Leouski and W. Croft. An evaluation of techniques for clustering search results. Technical Report IR-76, Department of Computer Science, University of Massachusetts, Amherst, 1996.

[37]. K. Lerman. Document clustering in reduced dimension vector space. Unpublished, 1999.

[38]. J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, pages 281–297, 1967.

Dr. A.Chandrabose "Improving Information Retrieval On The Web Using Clustering Approaches "'International Journal of Engineering Research and Applications (IJERA) , vol. 8, no.10, 2018, pp 01-08