International Journal of Applied Information Systems (IJAIS) ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 12 - No. 28, March 2020 - www.ijais.org

Luppar: Information Retrieval for Closed Text Document Collections

Fabiano Tavares da Silva, José Everardo Bessa Maia
State University of Ceará – UECE
60714-903 - Fortaleza – Ceará – Brazil

ABSTRACT

This article presents Luppar, an Information Retrieval tool for closed collections of text documents which uses a local distributional semantic model associated to each corpus. The system performs automatic query expansion using a combination of distributional semantic model and local context analysis and supports relevancy feedback. The performance of the system was evaluated in databases of different domains and presented results equal to or higher than those published in the literature.

General Terms:

Information Retrieval, Natural Language Processing

Keywords

Information Retrieval, Distributional Semantic Model, Local Context Analysis, Closed Document Collection

1. INTRODUCTION

The central issue in Information Retrieval (IR) is to address the need to user information, presented through a query, returning the documents relevant for this, if they exist. The Information Retrieval System (IRS), in general, have a simple interface: an input field, with free text to receive the query, and presents in return a list of the results considered relevant to this query. An example are the search engines.

The process is not always effective, and inefficiency is often caused by the inaccuracy with which a query formed by a few keywords models the actual user information need [5]. This causes the user to refine his query with various interactions making new words meaningful and then give up the search. A commonly used approach to solving this problem is through the query expansion [27] where new words or phrases are added to the query terms for IRS to retrieve new documents and approximate the need for user. The two problems commonly encountered in the initial query are of specificity or of meaning comprehension of the terms used.

Adding new terms to a query can be done manually by the user, either semi-automatically or automatically. In the manual form the user judges what could improve his query and reshapes the query terms. In the semiautomatic form the system assists the user, for example, by suggesting the addition of new terms. In automatic expansion there is no user participation, and the addition of new terms is implicitly performed. The Luppar approach performs automatic query expansion.

The automated query expansion (QE) in an IRS can be separated in two problems: the mechanism that triggers the expansion of the query and the method of expansion of the query. In relation to the mechanism that triggers the query expansion, can occur in three modalities, or by any of its combinations: expansion blind, based on the initial terms of the query; expansion by pseudo-relevance feedback, based on top documents retrieved by the initial query; or explicit relevance feedback expansion, in which the system interface provides the user with initial information about the retrieved documents together with links through which the user can indicate the documents, which are used in the expansion method.

The common idea of expansion methods is to construct or use a thesaurus to add new terms, assign weights, or recalculate them to modify the original representation. This process can be seen in the flowchart of Figure 1. The user's initial query is modified using terms from a thesaurus, which is then submitted to the search engine.

A thesaurus is a list of words with similar or related meanings. The thesaurus construction algorithm differentiates the approaches used. These algorithms can be characterized in three dimensions: by the scope of the texts used in the construction of the thesaurus, by the notion of context adopted (when using the hypothesis that the words meaning is given by their context of use) and by the measure used to evaluate the semantic similarity between words.

In relation to scope, a thesaurus may be broad, constructed for all vocabulary and known occurrences of the use of the words of a language, or may be restricted to terms present in a particular collection of documents. The notion of context refers to the hypothesis adopted on how to determine the meaning of words in the language. For example, one can adopt the hypothesis that the meaning of a word can be inferred from the words that occur in its surroundings, and that the adopted environment can be a certain window, the sentence, the whole paragraph or even the document . The measure of similarity, in turn, can vary by being probabilistic or deterministic, can consider or not the distance between words, or can still use external elements such as a domain ontology [4].

One of the best-known thesauri is Wordnet [20] constructed manually and with global characteristics. It has the advantage of bringing lexical information which solves problems of ambiguity in some cases. The disadvantages are that they are generic, so they do not bring gains in specific domains and are laborious to include new terms [22]. Even though it is difficult to grow this type of thesaurus several are the works exist [8, 17, 11] who use this approach to load pairs of synonyms in order to reshape queries.

Automatically constructed thesauri are based on the distributional hypothesis of [9] which states that words that are used and occur in the same contexts tend to have similar meanings. From this hypothesis, theories and methods were constructed to represent and quantify the similarity between linguistic items. A model based on this hypothesis is called Distributional Semantics. To create a representation, two types of models are usually constructed: counting models or predictive models. In the counting approach, the co-occurrence statistic of the words is traditionally used and thus vectors are created in word space [26]. The high dimensionality of this representation is then reduced, the densification being performed by singular value decomposition (SVD) [12] or by main component analysis (PCA) [13]. Predictive approaches use the term statistics to train neural networks that create dense vectors used as representations of terms [19]. For these representations one of the following measures of simi-

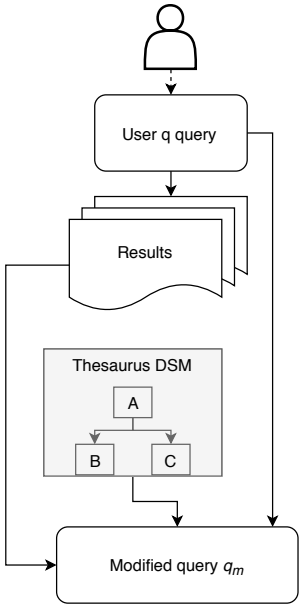

Fig. 1. Flow diagram for automatic query expansion used in Luppar. The results of the initial query are used together with a DSM thesaurus to generate the expanded query.

larity are often used: Cosine similarity [18], Lin measure [15] or Dice coefficient [6].
A closed collection of documents is understood as being a local electronic repository of documents, which is in the horizon of search and processing. This definition is in contrast to large-scale, multi-proprietary, and constantly evolving information retrieval on the Web. Examples of closed collections are a digital library or a repository of clinical patient information.
The Luppar proposal, described in this article, is a hybrid query expansion model combining local and global thesaurus properties. A thesaurus with global characteristics is constructed, in which a semantic distributional model is created for the terms of the vocabulary from the whole collection. It is combined with local context ideas by using only the top results from the initial query to constrain the context of the terms and increase the relevance of the retrieved documents. The remainder of this article is organized into three sections. Section 2, Methods, describes the algorithms used in Luppar. Section 3, Results, details of the implementation and evaluation procedures and results. Then finally the conclusion section.

2. MODELS

This section describes the concepts and methods used in the construction of Luppar. This is built on query expansion that uses a combination of Local Context Analysis with Distributional Semantic Model and a measure of associated text similarity.

2.1 Distributional Semantic Models (DSM)

This subsection briefly presents the ideas of DSM (Distributional Semantic Model) and how it is used in Luppar. The reader interested in a mathematical or algorithmic formulation is directed to [19], where *Word2vec* is precisely presented.
A fundamental characteristic of a thesaurus is the consistency of similarity between terms as measured by semantic similarity. The Distributional Semantic Models (DSM) are constructed using the distributional hypothesis of [9]. Similar context occurring words tend to have the same meaning. The DSM is a representation of words in geometric spaces of words (context) where vectors express concepts, and their proximity is a semantic measure. This means that words are semantically similar if the contexts (neighboring words) in which they appear are similar and should lead to their representations being close. Note that this is different from a representational space in which words are orthogonal to each other (bag of words).
Constructing a DSM involves the definition of a distribution model, formed by a quadruple [16]: the word vectors and their dimension; a function that takes into account the co-occurrences and how these items are represented in the vector; a function of similarity defined on vectors; and eventually, a mapping that transforms the vector space.
Given a corpus, it is possible to achieve a representation of words in a context space through a supervised training process. From this representation of words and their operations of similarity it is possible to construct a semantic thesaurus. Several studies show that word-based spaces (Word Embeddings) outperform traditional counting-based models to calculate word similarity [19]. This work used the Word2Vec predictive distributive model [19]. *Word2Vec* is a Word Embedding method used to induce vector space models using deep learning in neural networks with language models [19]. It is based on a simplified neural network with the number of entries proportional to the vocabulary words. The hidden layer realizes a linear projection with as many dimensions as the desired dimensionality of the vector space. This feature space is designed on a soft-max hierarchical output layer. The network is trained in each pair of input-output samples at a time, and for each pair the difference between the expected and the actual output of the network is calculated. The weights of the linear combination of the network are later adjusted to reduce the error using the back propagation procedure. This procedure is repeated for all training data pairs, often in several passages over the entire training data set, until the network converges and the error does not decrease further [3].
This method leads to good results because it has been shown to produce representations that preserve important linguistic characteristics [19]. In practice, [14] have shown that one of the main advantages of *Word2Vec* lies in its scalability, allowing training with up to billions of incoming text words in a few hours, differing from most others DSM.

2.2 Local Context Analysis with DSM

Automatic query expansion (without direct user interference) can be based on local or global methods. Local Analysis makes use of the best ranked documents captured by the initial query (first interaction), and without the user's participation, uses the closest terms to expand the query. On the other hand, the Global Analysis approach makes use of an external thesaurus, whether built by specialist or automatically. Already the work of Xu and Croft [27], it was proposed to use the first results of a query to construct a representation by competing concepts (groups of nouns) and by similarity of these with the query to find those candidates to be added to the query expansion, that is, to combine local and global analysis for QE. In Ermakova and Mothe [7] this process follows the same rationale but with refinement in the methods involved.
Algorithm 1 takes as input the original query q, and th, the DSM thesaurus previously constructed for closed collection of documents of the respective query. In line 1, the best ranked documents are retrieved with the original query. In lines 2 through 8, the n passages (text) best ranked in similarity to the original query are retrieved. This is achieved by breaking the documents initially retrieved by the query into passages (judgments) and classifying the passages as if they were documents. In lines 9 to 11, for each concept (noun group) in the top passages of the results, the similarity $sim(q, concepts[i], th)$ between the entire query q (not the individual query terms) and the concept is computed using a TF-IDF variant. In lines 12 and 13, finally, the m most well-ranked concepts, according to $sim(q, c)$, are added to the original query q. For each added concept a weight given by

Algorithm 1 Pseudocode for the QE proposed for IR.

Data: query q, thesaurus th
Result: modified query q_m
1: $documents \leftarrow$ search_top_ranked(q)
2: **for each** $documents$ **do**
3: $passages \leftarrow$ window($document$);
4: **for each** $passages$ **do**
5: $concepts \leftarrow$ find_concepts_in_context($passage$)
6: **end for**
7: **end for**
8: sort($concepts$)
9: **for** $i \leftarrow 1$ **to** N **do**
10: $m[i] \leftarrow$simqc($q, concepts[i], th$);
11: **end for**
12: sort(m)
13: $q_m \leftarrow q + m[1..n]$

$1-9x_i = m$, where i is the position of the concept in the ranking of concepts. The terms in the original query q can be emphasized by assigning a weight equal to 2 for each of them.
This algorithm for LCA query expansion has well defined local and global characteristics. The best of the local analysis is to use the top results that are assumed to be the best results. The global analysis use the concept of context and phrasal structures on the local set [2]. There are two sensitive points in algorithm 1. The first in line 5, which corresponds to the method of finding the concepts (group of more significant nouns at the top of the original query) and the second is the similarity function, in line 10, where the score between each concept and the query is calculated. It is precisely in these two aspects that our approach differs.
First, to extract the concepts a reduction of the representation of the documents of the rank top was used, which Croft [27] called passage, which is a reduced document structure with a window of size W. In our approach this window is automatically created using a complete period, until its closure by a syntactic signal. It was assumed that meanings are more closed to a complete period or paragraph. The period representation remains bag-of-word. Note that this strategy corresponds to a window of variable size.
The second sensible point of the algorithm is the similarity function that quantifies the correlation between the concept and a query term. The final goal is to expand the original query and estimate a P_q probability to be associated with it to be used in the language model to retrieve documents. In the work of Xu and Croft [27] it is proposed to use the simple correlation combined with a non-trivial variant of TF-IDF. In Ermakova and Mothe [7] work, besides making use of statistics about the document, information is also used of the terms that surround the terms of the query with a similarity formula that uses the distance and correlation.
The method in this work takes as a base the similarity proposed by Croft with the construction of short passages of the top of the rank to obtain the concepts, but also, like in the work of Emakorva, looks for to find words with meanings that will go besides the correlation between the terms of the query and the concepts. The strategy was to use the distributional hypothesis in function f to determine if the concept deserves to be a candidate term for expansion with strong meaning associated with query. Corpus defined off-line were used to build the DSM. These models are based on contexts that have similarity, and thus can associate an estimate of Pq to the contexts in which the query terms could appear in the documents.
In general terms, the method consists of using the passages to limit the universe of possibilities of terms that contextualize the query and this is one of the advantages of the local approach since the initial results bring much of the real intention of the user . The relevance of the query should now be refined for documents not perceived by questions of synonymy or polysemy. Then these passages are taken from the concepts according to LCA. These concepts are then selected according to the semantic similarity criterion constructed by the distributional model in a global way. This second step makes a total difference in comparison to the global thesaurus model because even though two words have semantic similarity and always appear in the same global context, it is through the LCA the relevance of the new term for expansion was restricted, for the collection of documents in focus.

2.3 Automatic Query Expansion (AQE)

The two-step recovery method proposed and implemented in Luppar makes use of similarity measures in two moments: in the initial recovery without expansion and in the recovery with the expanded query based on the documents ranking on top of those recovered in the first stage. A collection D of documents is given where each document d is represented in a vector space of words of t terms, with the terms indexed forming a vocabulary $V = k_1, k_2...k_t$. In this space each document is represented by a vector of weights of the words $q = \{w_{1,0}, w_{2,0}..., w_{t,0}\}$. Let q be the query represented as a pseudo document also in the same space q being w_t the weight associated with the term in the query. The similarity between query q and document d is expressed as follows:

$$sim(d, q) = \sum_{t \epsilon q \cap d} w_{t,d} \cdot w_{t,q} \ , \qquad (1)$$

where $w_{t,q}$ and $w_{t,d}$ are the weights calculated by the TF-IDF (frequency of the term by the inverse of the frequency in the documents): $w_{i,j} = (1 + \log f_{i,j}) \times \log \frac{N}{n_i}$, if $f_{i,j} > 0$, else 0.
The second step is the recovery with the query expanded. The AQE process is to use the terms of q to find new terms, without user input, to solve disambiguation problems and to discriminate documents better. Given initially that q will have new terms and will be called q'. The new terms compose w_j in q'. Now the similarity is calculated according to formula 1 but with $sim(d, q')$. The choice of the new terms is made by two thesauri. A global and static *WordNet* thesaurus [20] and another one constructed automatically using Distributed Semantic Model with Word Embedding representation [19].
In the online process the query has the same representation of the documents. The Vector Space Model (VSM) [25] and the probabilistic Okapi BM25 [24] were implemented and used as IR search models. In LCA [27] every process for expanding the query occurs online and does not depend on the global thesaurus. However, this approach requires the use of DSM, which requires pre-training to construct the model based on contexts. In the search process the method has as input the original query q that will be expanded and qualified with the probability P_q for each term of the original query and for the terms added to it. Starting with q a quick query with VSM is performed and get n documents at the top of the rank. Using these documents are created the passages (smaller and meaningful structures).
In this work a passage is a period closed by a syntactic mark. Passages are ordered as if they were small documents using VSM in the same way as with the original query. From these passages the m most well-ranked passages are selected and then extract the concepts that will serve as candidates for expansion. Then the best concepts (greater P_q) extracted from the top passages of the documents returned from the query are selected. The similarity $sim(q, c)$ between each concept c and the original query is computed by:

$$sim(q, c) = \prod_{k_i \in q}^{c} (\delta + \frac{\log(f(c, k_i) \times IDF_c)}{\log n})^{IDF_i} \ , \qquad (2)$$

where k_i corresponds to each term of q. IDF$_i$ and IDF$_c$ are the inverse of the frequency over term of the query i and about the

Table 1. Information about the datasets.

Base	Subject (Idiom)	N^o of Docs	N^o of Topics
MED	Medicine (EN)	1.033	30
LISA	Library (EN)	6.004	35
NPL	Elect Eng (EN)	11.429	100

concept c respectively. The $F(c, ki) = w2v.sim(c, k_i)$ where $w2v.sim$ is the function of Word2vect used to measure semantic similarity across the cosine of the vector generated by DSM. Next, the concepts that are closer to the query as a whole are ranked. The best concepts are chosen to be quantified according to their importance. In Ermakova and Mothe [7], propose to penalize candidates in various aspects (IDF, score, importance and POS), while this approach kept penalizing with weight 2 for the words of the original query and penalizing the concepts with $1 - 9xi = m$ [27] and also with the IDF, considering that the other scores did not significantly alter the accuracy.
Thus, in the application of the approach two macro processes are executed. The first off-line for corpus preprocessing, inverted index construction, storage of DSM counting statistics and training. The second is an online process responsible for effectively expanding the query by LCA and thus performing the search using the language models.
The project supports managing multiple document corpus. In the first step for each corpus is constructed an inverted index. In the index some statistics are stored, the language model and the dictionary of the terms. The terms in stopwords are removed. The Porter radicalization process [23] is applied. Frequency and distance are stored in the inverted index. In the process of constructing the document representation, the unigram (bag-of-word) model with weight factor TF-IDF is used for the relationship between terms and documents. TF-IDF uses the L2 frequency normalization [1]. With the complete unigram model we started the DSM training training phase. A predicted language model (Word Embedding) was used. Specifically the Word2Vect implementation [19] with CBOW (Continuos Bag-of-Word) architecture was used. The dimensionality of the vector used was size of 300. Window of size $W = 5$ for the contexts. With all these steps completed the system is ready to receive an inquiry.

3. RESULTS AND DISCUSSION

This section presents the data sets, merit indices used in the assessment, and the results.

3.1 Data and Metrics

The evaluation made use of three sets of data already commonly used as reference for evaluation in information retrieval. The three test collections were developed by Ed Fox at the Virginia Polytechnic Institute and State University [21]. Table 1 shows the characteristics of these collections.
All collections are made up of three files. A file with the documents, the second with the queries and the third the identification of the query and which documents were judged relevant by the consultation. These documents are entries for the proposed algorithms and its implementation produced a fourth document with the retrieval of recovered files. This organization known in the Cranfield paradigm allowed to evaluate and compare the need for information-document.
The MED data are short documents but with high accuracy queries without query expansion and with short queries. The LISA are documents with precision low and with long queries that will oppose the need to expand query. The NPL with short document size, but with a much larger documents, with medium precision and with many queries varying between long and short. In this way it was possible to trace the results on top of these three scenarios and to verify the best response to the approach.

To evaluate the performance of the ACL query expansion using DSM, implemented three other methods for comparison. The query expansion using to WordNet [20], that is, a global external thesaurus, the LCA itself with Local characteristics [27], but without using the DSM and the original without expansion, taken as baseline. The metrics used are the same as the TREC [10]. Even the same software called $trec\_eval$ in its last version 9.0. Of the eleven metrics produced, four were chosen for this job:

—**MAP:** precision on all queries;
—**Bpref:** calculates a preference relation, that is, if the documents judged are recovered before those deemed irrelevant;
—**Reciprocal Rank:** accuracy in relation to the first results.
—**The Recall-Precision curve:** 11 points interpolating the average precision (in 0%, 10%, ..., 100% of the recall) that allows to draw the coverage curve and precision allowing the realization that ‘as the most relevant documents are recovered (the recall increases) and while irrelevant documents are recovered (the accuracy decreases).

3.2 Results

Tables 2, 3 and 5 present the results of the experiments for the MED, LISA and NPL data sets. Each table shows four result columns: baseline, wordnet, LCA, and LCA-DSM. The baseline column refers to the retrieval of information by a query without expansion. The results in the wordnet column refer to the query expansion with the wordnet thesaurus. The LCA column records the results where the query expansion is based on the local context using Wordnet synonyms. Finally, the LCA-DSM column applies the word embedding combined with Local Context Analysis to construct a synonym dictionary and uses this synonym dictionary in the query expansion. The experiments were performed using the VSM and BM25 models.
Note that the results for LCA-DSM are consistently higher for the three bases, for the two models, and for the three performance indices. In only one case, the map metric, in model BM25 on the LISA database the index was slightly lower. The Recall-Precision curves in Figures 2, 4 and 3, one of the main benchmarks of performance for presenting a non-punctual comparison, confirm these results. In all of them, the curve for BM25-LCA-DSM is above the others. Note also that MAP is an approximate measure of the area under the non-interpolated recall-precision curve (RPAUC) and confirms these conclusions.

Table 2. Performance metrics results for MED collection.

Model	Metric	*Bline*	*WNet*	LCA	+DSM
VSM	map	0,5142	0,4949	0,5255	0,5348
	bpref	0,8985	0,9318	0,9418	0,9406
	RR	0,8537	0,7726	0,8889	0,8889
BM25	map	0,5033	0,4873	0,5262	0,5459
	bpref	0,8985	0,9318	0,9660	0,9712
	RR	0,8992	0,8294	0,8253	0,8944

Table 3. Performance metrics results for LISA collection.

Model	Metric	*Bline*	*WNet*	LCA	+DSM
VSM	map	0,2641	0,2034	0,2475	0,2602
	bpref	0,9981	1,0	1,0	0,9981
	RR	0,5184	0,4618	0,5006	0,5038
BM25	map	0,3495	0,2520	0,3577	0,3627
	bpref	0,9981	1,0	1,0	0,9981
	RR	0,6459	0.5085	0,6400	0,6693

Table 4. Performance metrics results for NPL collection.

Model	metric	*Bline*	*WNet*	LCA	+DSM
VSM	map	0,1886	0,1428	0,1968	0,2282
	bpref	0,9767	0,9886	0,9878	0,9333
	RR	0,4437	0,3583	0,5014	0,4267
BM25	map	0,2124	0,1756	0,2640	0,2580
	bpref	0,9766	0,9819	0,9891	0,9815
	RR	0,5987	0,5432	0,6142	0,6373

Table 5. Results for NPL collection

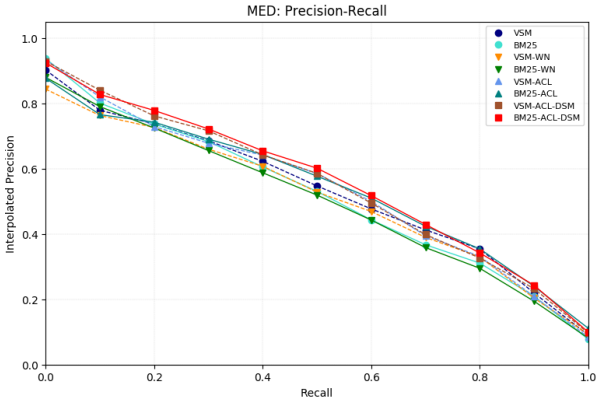

Fig. 2. Precision-recall curves for the MED data set showing that the performance of the BM25-ACL-DSM method is superior to the others compared.

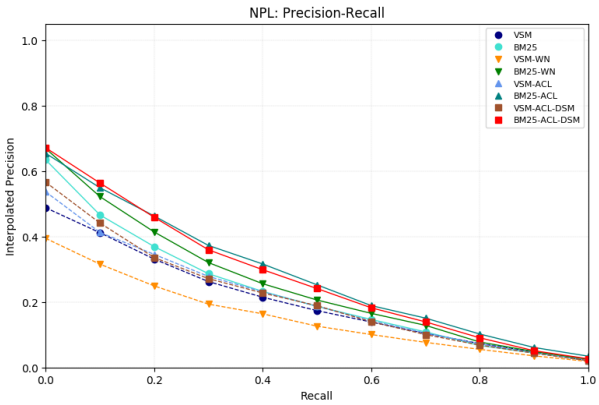

Fig. 3. Precision-recall curves for the NPL data set. The BM25 methods outperform the others. The Local Context Analysis (LCA) provides an additional gain.

4. CONCLUSION

Luppar is an Information Retrieval System (IRS) designed and implemented for corporate use, ie for closed text documents collections (not for web). The application takes advantage of this including a semantic thesaurus based on word embedding built only with the documents from the target collections. This approach prevents queries from being expanded with terms that, although they are meaningful in the language, are non-existent in the collections in focus. Word embedding restricted to the corpus combined with Local Context Analysis (LCA-DSM) complete the proposal in Luppar.
The paper used criteria and methods of the TREC conference to evaluate the proposal. The results of Tables 3, 2 and 4 show that the IR methods of Luppar are satisfactory and compatible with the state of the art. The experiments were constructed in order to reveal the efficiency gain of the LCA-DSM combination in

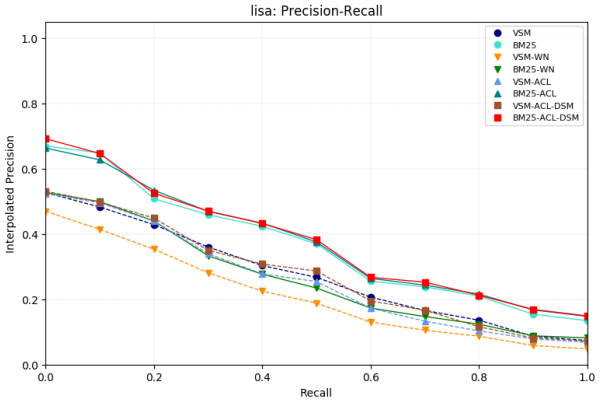

Fig. 4. Precision-recall curves for the Lisa data set. For this data set, the gain obtained with the Local Context Analysis (LCA) was lower.

relation to each method, LCA or DSM, applied in isolation. The IRS performance with the non-expansion query was included as the baseline method for control of the experiments.
Future work will be focused on putting Luppar into real application in a clinical case repository and news repository. Other evolution of this work extends it to retrieving information on the web. Luppar for web search is an ongoing project.

5. REFERENCES

[1] Gianni Amati and Cornelis Joost Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst.*, 20(4):357–389, October 2002.

[2] R. Baeza-Yates and B. Ribeiro-Neto. *Recuperação de Informação - 2ed: Conceitos e Tecnologia das Máquinas de Busca*. Bookman Editora, 2013.

[3] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

[4] J. Bhogal, A. Macfarlane, and P. Smith. A review of ontology based query expansion. *Inf. Process. Manage.*, 43(4):866–886, July 2007.

[5] Claudio Carpineto and Giovanni Romano. A survey of automatic query expansion in information retrieval. *ACM Comput. Surv.*, 44(1):1:1–1:50, January 2012.

[6] James R Curran and Marc Moens. Improvements in automatic thesaurus extraction. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition-Volume 9*, pages 59–66. Association for Computational Linguistics, 2002.

[7] Liana Ermakova and Josiane Mothe. Query expansion by local context analysis. In *Conference francophone en Recherche d'Information et Applications (CORIA 2016)*, pages pp–235, 2016.

[8] Zhiguo Gong, Chan Wa Cheang, and U Leong Hou. Web query expansion by wordnet. In *International Conference on Database and Expert Systems Applications*, pages 166–175. Springer, 2005.

[9] Zellig S Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

[10] Seyyed Hadi Hashemi, Charles LA Clarke, Jaap Kamps, Julia Kiseleva, and Ellen M Voorhees. Overview of the trec 2016 contextual suggestion track. In *Proceedings of TREC*, volume 2016, 2016.

[11] Ming-Hung Hsu, Ming-Feng Tsai, and Hsin-Hsi Chen. Query expansion with conceptnet and wordnet: An intrinsic comparison. In *Asia Information Retrieval Symposium*, pages 1–13. Springer, 2006.

[12] Thomas K Landauer and Susan T Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2):211, 1997.

[13] Rémi Lebret and Ronan Collobert. Rehabilitation of count-based models for word vector representations. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 417–429. Springer, 2015.

[14] Omer Levy and Yoav Goldberg. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the eighteenth conference on computational natural language learning*, pages 171–180, 2014.

[15] Dekang Lin. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics-Volume 2*, pages 768–774. Association for Computational Linguistics, 1998.

[16] Will Lowe. Towards a theory of semantic space. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 23, 2001.

[17] Meili Lu, Xiaobing Sun, Shaowei Wang, David Lo, and Yucong Duan. Query expansion via wordnet for effective code search. In *Software Analysis, Evolution and Reengineering (SANER), 2015 IEEE 22nd International Conference on*, pages 545–549. IEEE, 2015.

[18] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[19] Tomas Mikolov, Greg Corrado, Kai Chen, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *Proceedings of the International Conference on Learning Representations (ICLR 2013)*, 2013.

[20] George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November 1995.

[21] Virginia Disc One. Cd-rom from virginia polytechnic institute and state university. *Blacksburg, VA*, 1990.

[22] Jessie Ooi, Xiuqin Ma, Hongwu Qin, and Siau Chuin Liew. A survey of query expansion, query suggestion and query refinement techniques. *2015 4th International Conference on Software Engineering and Computer Systems, ICSECS 2015: Virtuous Software Solutions for Big Data*, pages 112–117, 2015.

[23] Martin F Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[24] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, April 2009.

[25] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, November 1975.

[26] Peter D Turney and Patrick Pantel. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188, 2010.

[27] Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '96, pages 4–11, New York, NY, USA, 1996. ACM.