Revisiting the Dependence Language Model for Information Retrieval

Loïc Maisonnasse
LIG-UJF
38041 Grenoble, France.
loic.maisonnasse@imag.fr

Eric Gaussier
LIG-UJF
38041 Grenoble, France.
eric.gaussier@imag.fr

Jean-Pierre Chevallet
IPAL-I2R
Singapore 119613
viscjp@i2r.a-star.edu.sg

ABSTRACT

In this paper, we revisit the dependence language model for information retrieval proposed in [1], and show that this model is deficient from a theoretical point of view. We then propose a new model, well founded theoretically, for integrating dependencies between terms in the language model. This new model is simpler, yet more general, than the one proposed in [1], and yields similar results in our experiments, on both syntactic and semantic dependencies.

Categories and Subject Descriptors: B.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

General Terms: Algorithms, Theory

Keywords: Information retrieval, language model, syntactic/semantic

1. DEPENDENCE LANGUAGE MODELS

[1] introduces a dependence language model (that we will refer to as *DLM*) for IR which, based on the standard language model ([5]), integrates syntactic dependencies in the computation of document relevance scores. This model relies on a variable L, loosely defined as a "linkage" over query terms, which is generated from a document according to $P(L|M_d)$, where M_d represents a document model. The query is then generated given L and M_d, according to $P(Q|L, M_d)$. In principle, the probability of the query, $P(Q|M_d)$, is to be calculated over all linkages Ls, but, for efficiency reasons, the authors make the standard assumption that these linkages are dominated by a single one, the most probable one: $L = \text{argmax}_L P(L|Q)$. $P(Q|M_d)$ is then formulated as:

$$P(Q|M_d) = P(L|M_d)\, P(Q|L, M_d) \quad (1)$$

In the case of a dependcy parser, as the one used in [1], each term has exactly one governor in each linkage L, so that the above quantity can be further decomposed, leading to:

$$\log P(Q|M_d) = \log P(L|M_d) + \sum_{i=1..n} \log P(q_i|M_d) + \sum_{(i,j)\in L} MI(q_i, q_j|L, M_d) \quad (2)$$

where MI denotes the mutual information, and:

$$P(L|M_d) \propto \prod_{(i,j)\in L} \hat{P}(R|q_i, q_j) \quad (3)$$

$\hat{P}(R|q_i, q_j)$ in the above equation represents the empirical estimate of the probability that concepts q_i and q_j are related through a parse in document d.

As the reader may have noticed, there is a certain ambiguity in the way the linkage L is used in the *DLM* model, ambiguity which is due, we believe, to the lack of a clear definition of what a linkage represents. In particular, according to equation 3, the probability $P(L|M_d)$ assumes the knowledge of the query terms, so that a linkage represents a set of dependencies over a set of known terms. However, in equation 1, such an interpretation cannot hold, as it would lead to disregard the term $P(Q|L, M_d)$ (as all the query terms are known in L), a quantity which is nevertheless necessary to derive the final form of the model given in equation 2. This ambiguity in the definition of L might not be important in practice, as it finally amounts to rely twice on the contribution of term pairs, which can be counter-balanced with appropriate smoothing. However, it is not completely satisfactory from a theoretical point of view.

Without loss of generality, we assume that a syntactic and/or semantic analysis of a query q can be represented as a graph $G_q = (C, E)$, where C is the set of terms (or concepts) in q, and E is a binary relation from $C \times C$ in $0, 1$ ($E(c_i, c_j) = 1$ if c_i and c_j are related, and 0 otherwise). The probability that the graph of query q is generated by the model of document d can be decomposed as[1]:

$$P(G_q|M_d) = P(C|M_d)\, P(E|C, M_d) \quad (4)$$

Assuming that, conditioned on M_d, query terms/concepts are independent of one another (a standard asumption in the language model), and that, conditioned on M_d and C, edges are independent of one another (again a standard assumption, also made in *DLM*), we can write:

$$P(C|M_d) = \prod_{c_i\in C} P(c_i|M_d) \quad (5)$$

$$P(E|C, M_d) = \prod_{(i,j)} P(E(q_i, q_j)|C, M_d) \quad (6)$$

[1]In the *DLM* model, a query is also implicitly represented as a graph (in fact a dependency parse), as the only linkage used is the most probable one, obtained by a parser trained on the collection.

SIGIR'07, July 23–27, 2007, Amsterdam, The Netherlands.
ACM 978-1-59593-597-7/07/0007.

	syntactic dependencies				semantic relations			
	λ_u	λ_r	training	test	lambda_u	lambda_r	training	test
MAP								
DLM	0.8	0.9	0.194	0.210	0.1	0.9	0.256	0.333
GLM	0.4	0.9	0.218	0.209	0.1	0.1	0.243	0.337
LM	0.4		0.220	0.209	0.1		0.255	0.339
P@5								
DLM	0.2	0.9	0.288	0.287	0.1	0.9	0.450	0.440
GLM	0.2	0.9	0.344	0.300	0.1	0.1	0.433	0.480
LM	0.2		0.336	0.287	0.1		0.408	0.453

Table 1: Mean average precision and precision af 5 documents for the *DLM*, *GLM* and *LM* models. Models are trained on 25 queries and evaluated on 30 queries from ImageCLEFmed 2005 and 2006. λ_u and λ_r correspond to the smoothing parameters for the unigrams and relations (contribution from the collection).

Equation 5 corresponds to the standard language model (potentially applied to concepts), and is similar to the second contribution of the right-hand term of equation 2. The quantities $P(E(q_i, q_j)|C, M_d)$ of equation 6 can be directly estimated through maximum likelihood. Following standard practice in language modeling, one can furthermore "smooth" this estimate by adding a contribution from the collection. This results in:

$$P(E(c_i, c_j) = x|C, M_d) = (1-\lambda)\frac{xD(c_i, c_j, \mathcal{R}) + (1-x)D(c_i, c_j, \neg\mathcal{R})}{D(c_i, c_j, \mathcal{R}) + D(c_i, c_j, \neg\mathcal{R})} + \lambda\frac{xC(c_i, c_j, \mathcal{R}) + (1-x)C(c_i, c_j, \neg\mathcal{R})}{C(c_i, c_j, \mathcal{R}) + C(c_i, c_j, \neg\mathcal{R})}$$

where $D(c_i, c_j, \mathcal{R})$ ($C(c_i, c_j, \mathcal{R})$) is the number of times c_i and c_j are linked in the document (collection). Similarly, $D(c_i, c_j, \neg\mathcal{R})$ ($C(c_i, c_j, \neg\mathcal{R})$) is the number of times c_i and c_j are observed together in the document without being linked.

The model we have just defined (which we will refer to as *GLM*, for *Graph Language Model*) is well motivated from a theoretical point of view, and can be applied to any graphical representation of queries and documents. Furthermore, it relies on only two terms, which are easy to estimate, whereas the *DLM* model uses three terms, with a somewhat complex estimation of the term $P(L|M_d)$. Lastly, it is easy to see that the *GLM* model generalizes the bigram model presented in [6]. We now show how this model behaves experimentally.

2. EXPERIMENTAL VALIDATION

In order to assess the *GLM* model and answer the above question, we conducted two series of experiments on the collection of ImageCLEF[2]. In the first series, we used MiniPar ([3]) to produce a dependency parse for both queries and documents. In the second series, we derived a semantic graph for queries and documents from UMLS. In the latter case, we replaced all possible instances of concepts by their corresponding concept(s), and retained all the relations between concepts provided in the semantic network associated with UMLS. As the collection in ImageCLEF focuses on pathologies and anatomic diseases, we did not take into account the NCI and PDQ thesauri of UMLS, which focus on cancer. This filtering step is similar to the one proposed in [4]. In all cases, we retained only full words (ie words corresponding to nouns, adjectives, verbs abd adverbs). In order to estimate the parameters of our models (namely the smoothing coefficients), we divided the 55 queries available from ImageCLEF 2005 and 2006 in two sets: 25 queries were randomly selected for training, and 30 for testing. Lastly, we retained two measures for evaluation: the mean average precision, and the precision at 5 documents. Also note that we use the *DLM* model *as is* on semantic relations, even though this use is not theoretically justified.

Table 1 shows that on both the syntactic and semantic dependencies, the models *DLM* and *GLM* performs in a similar way (no significant difference was detected using a Wilcoxon signed rank test at the level 0.05). On this collection, there is furthermore no significant difference between these two models and the *LM* model, which does not make use of the relations between terms. This observation agreees with some of the results reported in eg [2]. Lastly, it is interesting to note that the semantic indexing we have retained significantly improves the results over the syntactic one, and shoud be preferred here.

[2]http://ir.shef.ac.uk/imageclef/. This collection consists of written diagnostic with associated images. We retrieved only the text part of documents as it enables the use of semantic relations presents in UMLS and thus allows testing the integration of syntactic and semantic graphs.

3. CONCLUSION

In this paper, we have shown that the *DLM* proposed in [1] is flawed theoretically. We have then proposed a new model for integrating dependencies between terms that is (a) well founded theoretically, (b) simpler and (c) more general. Our experimental results suggests that this new, simpler model behave similarly to the *DLM* model, and may thus be preferred over it.

4. REFERENCES

[1] J. Gao, J.-Y. Nie, G. Wu, and G. Cao. Dependence language model for information retrieval. In *SIGIR*, 2004.

[2] C. Lee, G. G. Lee, and M. G. Jang. Dependency structure language model for information retrieval. In *ETRI journal*, 2006.

[3] D. Lin. Dependency-based evaluation of MiniPar. In *Workshop on the Evaluation of Parsing Systems*, 1998.

[4] Y. H. H. Lowe and W. Hersh. A pilot study of contextual UMLS indexing to improve the precision of concept-based representation in XML-structured clinical radiology reports. In *AMIA*, 2003.

[5] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR*, 1998.

[6] M. Srikanth and R. Srikanth. Biterm language models for document retrieval. In *SIGIR*, 2002.