

# Inferring Document Similarity from Hyperlinks

David Grangier

Samy Bengio

IDIAP Research Institute  
Rue du Simplon 4, CP 592  
1920 Martigny, Switzerland  
{grangier,bengio}@idiap.ch

## ABSTRACT

Assessing semantic similarity between text documents is a crucial aspect in Information Retrieval systems. In this work, we propose to use hyperlink information to derive a similarity measure that can then be applied to compare any text documents, with or without hyperlinks. As linked documents are generally semantically closer than unlinked documents, we use a training corpus with hyperlinks to infer a function  $a, b \rightarrow \text{sim}(a, b)$  that assigns a higher value to linked documents than to unlinked ones. Two sets of experiments on different corpora show that this function compares favorably with *OKAPI* matching on document retrieval tasks.

### Categories and Subject Descriptors:

H.3.3 [Information Storage and Retrieval]: Miscellaneous  
I.2.6 [Artificial Intelligence]: Learning

**General Terms:** Algorithms, Experimentation

**Keywords:** hyperlinks, similarity measure, matching measure, term weighting, gradient descent, neural networks

## 1. INTRODUCTION

Automatic techniques to access and organize document collections are essential to fully benefit from large text corpora. Several of these methods require a measure to quantify semantic similarities between text items: e.g. clustering relies on document comparisons, while Information Retrieval (IR) depends on document/query similarities.

In this work, our goal is to infer a measure of similarity relying on the semantic relationships contained in a hyperlinked corpus. In such a corpus, links can be considered as indicators of topic relatedness, i.e. linked documents tend to be semantically closer than unlinked documents [1]. Therefore, we propose to identify a measure of similarity  $a, b \rightarrow \text{sim}(a, b)$  such that, for any document  $d$ , the documents which are linked to it are considered more similar than those which are not:

$$\forall d \in D_{\text{train}}, \forall l^+ \in L(d), \forall l^- \notin L(d), \text{sim}(d, l^+) > \text{sim}(d, l^-) \quad (1)$$

where  $L(d)$  is the set of documents linked with  $d$ . For that purpose, a gradient descent strategy [2] is adopted: we first introduce a parameterized measure of similarity  $a, b \rightarrow \text{sim}_\theta(a, b)$  and a cost  $C$  which indicates how far  $\text{sim}_\theta$  is

from the condition (1), then gradient descent optimization is used to select the parameters  $\theta^*$  which minimize  $C$  for a given training corpus  $D_{\text{train}}$ .

The inferred measure  $\text{sim}_{\theta^*}$  can then be applied to any pair of text documents, with or without hyperlinks, in any context where a text similarity measure is needed. In order to evaluate this approach, we compared the inferred measure with the state-of-the-art *OKAPI* matching measure [3] over two retrieval tasks (see Section 3). In this context, our model *LinkLearn* is shown to improve both precision at top 10 and average precision with respect to *OKAPI*.

In the remainder of this paper, Section 2 describes the proposed method, Section 3 presents the experiments and results, and Section 4 draws some conclusions.

## 2. THE LINKLEARN MODEL

This section describes the two main parts of the *LinkLearn* Model: the parameterized measure of similarity  $\text{sim}_\theta$  is first defined and the cost  $C$  related to condition (1) is then introduced.

### 2.1 Model Parameterization

Our model relies on the Vector Space Model (VSM): each document  $d$  is represented with a vector  $(d_1, \dots, d_V)$ ,  $V$  being the vocabulary size, and the documents are then compared according to the inner product of their vectors,

$$\text{sim}(d, d') = \sum_{i=1}^V d_i \cdot d'_i.$$

The weight  $d_i$  of a term  $i$  in a document  $d$  is a function of  $tf_{d,i}$  (the number of occurrences of  $i$  in  $d$ ),  $idf_i$  (the inverse document frequency of  $i$ ) and  $nll_d$  (the length of document  $d$  divided by the average document length):

$$d_i = f(tf_{d,i}, idf_i, nll_d).$$

This choice is motivated by the fact that functions of those three variables have led to the best performances in TREC IR benchmarks<sup>1</sup>, *OKAPI BM25* being the most used of those functions:

$$d_i^{OKAPI} = \frac{(K+1) \cdot tf_{d,i} \cdot idf_i}{K \cdot ((1-B) + B \cdot nll_d) + tf_{d,i}},$$

where  $K$  and  $B$  are hyperparameters. In our case, the function  $g$  is chosen to be the product of three Multi-Layer Perceptron (MLP) functions:

$$\begin{aligned} d_i &= f(tf_{d,i}, idf_i, nll_d) \\ &= MLP_{tf}(tf_{d,i}) \cdot MLP_{idf}(idf_i) \cdot MLP_{nll}(nll_d). \end{aligned} \quad (2)$$

This parameterization makes the simplifying assumption that  $tf_{d,i}$ ,  $idf_i$  and  $ndl_d$  variables are independent. Such a hypothesis improves greatly the model efficiency (see [4] for further explanations) while still allowing for good performance (see Section 3).

## 2.2 Similarity Constraint Criterion

As mentioned above, it is desirable that, for any document  $d$ , the documents which are linked to it (i.e. the documents of  $L(d)$ ) are considered more similar to  $d$  than any other documents (1). A simple cost would hence be the proportion of document triplet  $d \in D_{train}$ ,  $l^+ \in L(d)$ ,  $l^- \notin L(d)$  for which the above property is not satisfied:

$$C^{0/1} = \frac{1}{|D_{train}|} \sum_{d \in D_{train}} C_d^{0/1} \quad (3)$$

where

$$C_d^{0/1} = \frac{1}{|L(d)| \cdot |L(d)|} \sum_{\substack{l^+ \in L(d) \\ l^- \in L(d)}} I\{sim(d, l^+) < sim(d, l^-)\},$$

$I\{\cdot\}$  is the indicator function, i.e.  $I\{c\} = 1$  if  $c$  is true and zero otherwise and  $L(d)$  is the set of documents linked with  $d$  (i.e. the documents referring to  $d$  and the documents referred to by  $d$ ).

Similarly to the 0/1 loss (i.e. error rate) in the case of classification,  $C^{0/1}$  cannot be directly minimized through gradient descent [2]. Hence, we propose to minimize an upper bound of this quantity:

$$C = \frac{1}{|D_{train}|} \sum_{d \in D_{train}} C_d \quad (4)$$

where

$$C_d = \frac{1}{|L(d)| \cdot |L(d)|} \sum_{\substack{l^+ \in L(d) \\ l^- \in L(d)}} \|1 - sim(d, l^+) + sim(d, l^-)\|_+,$$

and  $x \rightarrow \|x\|_+$  is 0 for  $x < 0$  and  $x$  otherwise.  $C$  is actually an upper bound of  $C^{0/1}$  since  $\forall x \in \mathbb{R}, I\{x < 0\} \leq \|1 - x\|_+$ . This function  $C$  is derivable almost everywhere and we can hence select the parameters of the MLPs (2) which minimize  $C$  through gradient descent [2] (see [4] for further details).

## 3. EXPERIMENTS AND RESULTS

The following section describes the two sets of retrieval experiments performed in order to assess the proposed method. In both cases, *LinkLearn* is compared with *OKAPI*.

### 3.1 Wikipedia Experiments

The experiments presented in the following are performed over the Wikipedia corpus [5]. This dataset consists of  $\sim 450,000$  encyclopedia articles, each article referring to other related articles using hyperlinks.

The corpus has been randomly split into 3 subsets of 150,625 documents: *train*, *valid* and *test*. The *train* set is used for gradient descent (i.e.  $C$  is minimized over this set) and *valid* is used to select the hyperparameters for both *LinkLearn* (the number of hidden units in the MLPs, the

Table 1: Wikipedia Results

	<i>OKAPI</i>	<i>LinkLearn</i>
Precision at top 10	21.5%	25.2% (+18%)
Break Even Point	36.6%	42.1% (+15%)
Average Precision	37.3%	43.8% (+17%)

Table 2: TDT-2 Results

	<i>OKAPI</i>	<i>LinkLearn</i>
Precision at top 10	38.8%	43.2% (+11%)
Break Even Point	30.3%	35.2% (+16%)
Average Precision	29.3%	34.5% (+18%)

number of training iterations and the learning rate of the gradient descent) and *OKAPI* ( $K$  and  $B$ ).

The *test* set is used only for evaluation, in which we perform a related document search: each document is considered to be a query whose relevant documents are the documents linked with  $d$ . Table 1 shows that, according to all performance measures, *LinkLearn* outperforms *OKAPI*.

### 3.2 TDT-2 Experiments

To have a more complete evaluation, we also compared *LinkLearn* and *OKAPI* matching measure on TREC queries for the TDT-2 corpus [6]. Without re-training or adaptation, the measure inferred from the hyperlinked Wikipedia data has been applied as a query/document matching measure to the non-hyperlinked TDT-2 corpus.

The results obtained over TDT-2 confirm those obtained over Wikipedia (see Table 2): the use of *LinkLearn* leads to an improvement with respect to *OKAPI* matching according to the different performance measures used.

## 4. CONCLUSIONS

In this paper, we introduced *LinkLearn*, a gradient descent approach to derive a document similarity measure from a hyperlinked training corpus: the measure is selected such that, in most cases, a document is considered more similar to the documents with which it is linked than to the other documents. This approach has shown to be effective in an IR context: the use of the similarity measure inferred by *LinkLearn* has led to higher retrieval performances when compared to the state-of-the-art *OKAPI* matching measure.

**Acknowledgments:** This work has been performed with the support of the Swiss NSF through the NCCR-IM2 project. It was also supported by the PASCAL European Network of Excellence, funded by the Swiss OFES.

## 5. REFERENCES

- [1] B. D. Davison, "Topical locality in the web," in *ACM Special Interest Group on Information Retrieval*, 2000.
- [2] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.
- [3] S. E. Robertson et al, "Okapi at TREC-3," in *NIST Text Retrieval Conference*, 1994.
- [4] David Grangier et al, "Inferring document similarity from hyperlinks," Tech. Rep. RR 05-21, IDIAP, 2005.
- [5] "Wikipedia, the free encyclopedia," [www.wikipedia.org](http://www.wikipedia.org).
- [6] C. Cieri et al, "The TDT-2 text and speech corpus," in *DARPA Broadcast News Workshop*, 1999.