A language model which integrates uncertainty

Caroline Tambellini
LIG Laboratory
BP 53
38041 Grenoble cedex 9 - France
caroline.tambellini@imag.fr

Abstract: All information retrieval models are based on the equality which exists between the query terms and the documents terms. This principle is based on the assumption that terms are correctly extracted from the documents and queries. But in some context (such as document generated by automatic speech recognition or by optical character recognition systems), this assumption is not true. We propose a generalisation of the information retrieval model based on the language model integrating this dimension. To do it, we introduce two notions: the term certainty value (in relation with the extraction process) and the pairing between two terms. Pairing between two terms is defined by their relative position (called the concordance) and the area they have in common (called the intersection).

Keywords: information retrieval model, uncertain data, term certainty value, pairing, concordance, intersection

1. INTRODUCTION

Usually, in information retrieval, a matching function is used to determine a document ranking from a corpus in response to a query. This function matches terms of a query with terms of a document determining if the terms of a query are present or not in a document. We can distinguish two dimensions: term presence ($t_{d,j} = t_{q,i}$) and term absence ($t_{d,j} \neq t_{q,i}$). To do it, we assume that terms used in document representation are equal to initial terms (i.e. terms really used in the document).

Ex: $Q = \{\text{'tomato'}\}$ and $D1 = \{\text{'tomato'}\}$, $D2 = \{\text{'tomato'}, \text{'orange'}\}$, $D3 = \{\text{'tamato'}\}$

Only the documents D1 and D2 which contain the term of the query Q are considered as relevant for the query Q. The document D3 is considered as non relevant even if his term 'tamato' is probably the term 'tomato' badly recognized.

In some contexts, the assumption described above is not completely true. For example, in the automatic speak recognition (ASR) or optical character recognition (OCR) domains, the system can have some transcriptions errors: badly recognized words, missed words or added words. It is what is called uncertain data. In such contexts, each term is associated to a certainty value provided by the extraction process.

Ex: $D1 = \{\text{'tomato'}, 0.2\}$, $D2 = \{\text{'tomato'}, 0.6, \text{'orange'}, 0.8\}$, $D3 = \{\text{'tamato'}, 0.5\}$. *If we take into account that 'tamato' is potentially the term 'tomato' badly recognized, all the documents are considered as relevant. The question is to know what document is the most relevant.*

The matching function takes into account if the terms of the query are present or absent in the document and we add a new case: the term of the query is almost present ($t_{d,j} = \sim t_{q,i}$) in the document.

We call the way two terms are associated the pairing. Pairing between two terms is defined by their relative position (called the concordance) and the area they have in common (called the intersection).

To take into account the three types of possible equality between a term of a query and a term of a document, we must integrate two notions: term certainty value and pairing (cf. Figure 1).

Figure 1. General principle

In this paper, we deal with this problem of uncertain data. First we present a state of the art of information retrieval system models performances with uncertain data. Then we define the pairing notion with the concordance and the intersection between two terms. After having described document and query representations, we develop our matching function. Finally, we present our first evaluations showing that the handling of the term certainty value in the term weighting improves results of retrieval.

2. STATE OF THE ART

Information retrieval applicability to uncertain data is a problem that was born with the expansion of the number of electronic documents from scanned “paper” documents or documents from audio recordings translated by automatic speech recognition system. Indeed, lots of documents were scanned to be available online. These documents from optical character recognition system are uncertain because of badly recognized words, missed words and added words. There is the same problem for the documents generated by automatic speech recognition systems.

Research has been done to compare performances of several information retrieval systems on uncertain data versus on correct data (i.e. data from manual transcription).

Taghva [9] compares performances of Boolean information retrieval system on OCR documents versus its corresponding correct documents (i.e. from manual transcription). Croft [3] does the same study on vectorial information retrieval system (SMART) and probabilistic information retrieval system (INQUERY); Mittendorf [5] do it also on probabilistic information retrieval system. Their conclusion is that the effects of uncertain data on information retrieval system performances are not significant with long documents: recall is not affected and average precision decreases by 3%. Average precision is more affected for short documents: - 10%.

Grangier [4] compares performances of probabilistic information retrieval system on ASR documents versus its corresponding correct documents. His conclusion is the same as the one given for OCR documents. The average precision for long documents decreases by 5%.

It is not surprising that results are almost the same for uncertain data (i.e. OCR or ASR documents) than for their corresponding correct documents because a long document allows a smoothing of errors. In fact, a same word can be sometimes badly recognized in the document and sometimes correctly recognized. So, globally the word is known as being in the document.

On the other way, short documents are not included in such experiments, even if the bad recognition of a word in such documents has a direct and bad influence on the quality measure (recall, precision). We propose in this paper to integrate not only the absence or presence of each term but also its approximation due to a probably bad extraction.

3. MATCHING PRINCIPLE

We want to introduce the uncertainty notion (i.e. certainty value and pairing) in the language model adapted to information retrieval. Usually, the language model principle is based on the probability $P(Q | \mathcal{M}_D)$ that a query Q is generated by a document model $\mathcal{M}_D$ [5] and the assumption of the independence of the terms of the query ($Q = t_{q,1}, t_{q,2}, \dots, t_{q,N(q)}$) is done [7]:

$$P(Q | \mathcal{D}) = \prod_{t_{q,i} \in Q} \lambda P(t_{q,i} | \mathcal{M}_D) + (1 - \lambda) P(t_{q,i} | \mathcal{M}_C)$$

$P(t_{q,i} | \mathcal{M}_C)$ is the probability of the term $t_{q,i}$ to be in the corpus of documents $\mathcal{C}$ (i.e. $t_{c,j} = t_{q,i}$). This probability is used in the matching function to avoid having null probability for the query because of a term absence (i.e. $t_{d,j} \neq t_{q,i}$) in the document: this is a smoothing of the language model.

This writing is based on the probability of a term in a document model. To take into account uncertain notion, we must add a new dimension: term approximation (i.e. $t_{d,j} = \sim t_{q,i}$). To do it, we add a third parameter in the matching function: $P(\sim t_{q,i} | \mathcal{M}_D)$. This is the probability of the term approximations in the document model $\mathcal{M}_D$.

$$P(Q | \mathcal{D}) = \prod_{t_{q,i} \in Q} \mu \lambda P(t_{q,i} | \mathcal{M}_D) + (1 - \lambda) P(t_{q,i} | \mathcal{M}_C) + (1 - \mu) P(\sim t_{q,i} | \mathcal{M}_D)$$

Before developing each component of the function, we define the concept of pairing.

4. PAIRING NOTION

Given two terms x , y of respective length n_x and n_y . These two terms can be paired together i.e. they are close to each other taking only into account their string value and their relative position, in a more or less strong way:

- according to their concordance, i.e. their relative position
- according to their intersection.

Figure 2. x and y start together and $n_x < n_y$

In the example of the Figure 2, we can measure pairing between χ and y by taking into account their concordance (their relative position), i.e. χ begins y : $Conc(\chi, y) = \text{begins}$ and by taking into account their intersection (shaded area) i.e. χ for the one and $y1$ for the other: $Inter(\chi, y) = (\chi, y1)$.

Example 1: $\chi = \text{'import'}$ and $y = \text{'importance'}$,
 'import' begins 'importance' and their intersection is 'import' for both.
In this case, we have $Conc(\chi, y) = \text{begins}$ and $Inter(\chi, y) = (\text{'import'}, \text{'import'})$.

Example 2: $\chi = \text{'tomato'}$ and $y = \text{'tamato'}$,
 'tomato' concurs 'tamato' and their intersection is 'tomato' for the one and 'tamato' for the other.
In this case, we have $Conc(\chi, y) = \text{concurs}$ and $Inter(\chi, y) = (\text{'tomato'}, \text{'tamato'})$.

To describe more precisely the type of equality existing between two terms, a typology of relations between two terms is necessary. Our concordance typology between two terms is based on Allen's relations. Allen's relations allow to define temporal position of all the document objects while placing temporal relations between them.

4.1. Concordance between two terms

Allen's model is a complete graph where the arcs express relations and the nodes represent the intervals [1]. This model defines thirteen possible relations between two intervals. There are relations of connexity (precedes, meets, overlaps), inclusion (begins, during, finishes), simultaneity (equal) and their opposite. If we transpose Allen's relations to our context, we get the typology illustrated by Figure 3. We now deal with concordance between two terms instead of temporal relations. Two terms can concord between them in different ways; this is what is called the concordance (cf. Figure 3, column 2).

Figure 3. Concordance and intersection between two terms

4.1.1. Nature of the concordance between two terms

All the concordances between two terms χ and y is defined by $\mathcal{A}$:

$\mathcal{A} = \{\text{begins}, \text{is_began_by}, \text{is_included_in}, \text{includes}, \text{ends}, \text{is_ended_by}, \text{overlaps}, \text{concurs}, \text{not_concurs}\}$.

We define the function $Conc(\chi, y)$ which determines the nature of the concordance between the terms χ and $y \in \mathcal{V}$ ($\mathcal{V}$ being a closed set of terms).

4.1.2. Value of the concordance

All the possible values of the concordance are defined by $\mathcal{ValConc}$.

$$\mathcal{ValConc}(a) = \begin{cases} \alpha_1 & \text{si } a = \textit{begins} \\ \alpha_2 & \text{si } a = \textit{is_began_by} \\ \alpha_3 & \text{si } a = \textit{is_included_in} \\ \alpha_4 & \text{si } a = \textit{includes} \\ \alpha_5 & \text{si } a = \textit{ends} \\ \alpha_6 & \text{si } a = \textit{is_ended_by} \\ \alpha_7 & \text{si } a = \textit{overlaps} \\ \alpha_8 & \text{si } a = \textit{concurr} \\ 0 & \text{si } a = \textit{not_concurr} \end{cases}$$

4.2. Intersection

Figure 3 (third column) shows intersection associated to each concordance (previously defined).

4.2.1. Nature of the intersection

The nature of an intersection between two terms χ and y is defined as:

$$\begin{aligned} \text{Inter} : C^{*2} &\rightarrow C^{*2} & (C \text{ being a closed set of characters.}) \\ (\chi, y) &\rightarrow (\chi', y') \end{aligned}$$

4.2.2. Intersection value

We define the intersection value between two terms $(\chi, y) \in \mathcal{V}^2$ by the function $\mathcal{ValInter}$:

$$\begin{aligned} \mathcal{ValInter} : \mathcal{V}^2 &\rightarrow [0, 1] \\ (\chi, y) &\rightarrow \beta \end{aligned} \quad \mathcal{ValInter}(\chi, y) = \begin{cases} 1 & \text{if } \chi = y \\ 0 \leq \mathcal{ValInter}(\chi, y) < 1 & \text{otherwise} \end{cases}$$

For example, we can have the following situation: $\text{Conc}(\chi, y) = \textit{concurr}$ and χ and y have a partial equivalence because we are working in a context where data are uncertain. This intersection equivalence degree between the two terms is expressed by $\mathcal{ValInter}(\chi, y)$.

Problem of uncertain data is known in several domains, such as determination of synonyms or in orthographic correction systems; the algorithms used for these domains can be also used to determine $\mathcal{ValInter}(\chi, y)$.

5. DOCUMENT REPRESENTATION

We describe the document model definition which integrates term certainty.

A document $\mathcal{D}$ is a sequence of length $\mathcal{N}_{\mathcal{D}}$ such as:

$$\mathcal{D} = [(t_{d,1}, c_1), (t_{d,2}, c_2), (t_{d,3}, c_3), \dots, (t_{d,\mathcal{N}_{\mathcal{D}}}, c_{\mathcal{N}_{\mathcal{D}}})] \text{ with } t_{d,i} \in \mathcal{V}, c_i \in \mathfrak{R}^+ \text{ and } c_i = \textit{Cert}(t_{d,i}).$$

5.1. Term certainty

A value of certainty c is associated for each term $\chi \in \mathcal{V}$. This value is dependent on extraction process and is represented by the function named $\textit{Cert}$.

5.2. Document model

We called $\mathcal{M}_{\mathcal{D}}$ the unigram model of the document $\mathcal{D}$.

$$\mathcal{M}_{\mathcal{D}} = \{(t_{d,i}, \mathcal{P}(t_{d,i})), \forall i \in [1 \dots \mathcal{N}_{\mathcal{D}}]\} \text{ with } t_{d,i} \in \mathcal{V} \text{ and } \mathcal{N}_{\mathcal{D}} \leq \mathcal{N}_{\mathcal{D}}$$

$$\mathcal{M}_{\mathcal{D}} = \{(t_{d,1}, \mathcal{P}(t_{d,1})), (t_{d,2}, \mathcal{P}(t_{d,2})), \dots, (t_{d,i}, \mathcal{P}(t_{d,i})), \dots, (t_{d,\mathcal{N}_{\mathcal{D}}}, \mathcal{P}(t_{d,\mathcal{N}_{\mathcal{D}}}))\} \text{ with } t_{d,i} \in \mathcal{V} \text{ and } \forall i, \forall j, i \neq j, t_{d,i} \neq t_{d,j}$$

Usually in a language model, a count of term occurrences is used ($\sum t_{d,i}$) to calculate the probability of the term $t_{d,i}$ in the document model:

$$\mathcal{P}(t_{d,i}) = \|t_{d,i}\| \text{ with } \|t_{d,i}\| = \frac{|t_{d,i}|}{|\textit{document}|} = \frac{|t_{d,i}|}{\mathcal{N}_{\mathcal{D}}}$$

In a context with terms associated with certainty value, the count is not made only on the number of occurrences but also on terms certainty values. The probability of a term in a document is dependent both on the maximum likelihood estimation and on the certitude of extraction. So, we have:

$$P(t_{d,i}) = \frac{\sum_{t_{d,i}} Cert(t_{d,i})}{\mathcal{N}_d} \leq \frac{|t_{d,i}|}{\mathcal{N}_d}$$

6. QUERY REPRESENTATION

A query Q is a sequence $\mathcal{N}_q$ length :

$$Q = [(t_{q,1}, c_1), (t_{q,2}, c_2), (t_{q,3}, c_3), \dots, (t_{q,\mathcal{N}_q}, c_{\mathcal{N}_q})] \text{ with } t_{q,i} \in \mathcal{V} \text{ and } c_i \in \mathbb{R}^+.$$

7. MATCHING FUNCTION

7.1. Definitions

As in most of the language models, the following simplification was made: we assume the independence between terms in the query.

The matching function associated to the language model takes into account the case where the term of the query is present in the document and the case where the term is absent in a document but present in the corpus (thanks to a smoothing function). In our working context, we take into account three dimensions: presence of the term in the document, presence of the term in the corpus and presence of approximations of a term in the document.

To define matching function, we used the principle of the query generation by the document to evaluate document score in response to a query. With a query Q and a document $\mathcal{D}$ as defined previously, we have the following matching function (cf. 3. Matching principle):

$$P(Q | \mathcal{D}) = \prod_{t_{q,i} \in Q} \mu \lambda P(t_{q,i} | \mathcal{M}_\mathcal{D}) + \mu (1 - \lambda) P(t_{q,i} | \mathcal{M}_\mathcal{C}) + (1 - \mu) P(\sim t_{q,i} | \mathcal{M}_\mathcal{D})$$

7.1.1. The term is present in the document :

$$Presence : CI \times \mathcal{V} \rightarrow \mathcal{V}$$

$$(\mathcal{M}_\mathcal{D}, \chi) \rightarrow t_j \in \mathcal{M}_\mathcal{D} \mid Conc(\chi, t_j) = Concurs \quad \text{and} \quad ValInter(Inter(\chi, t_j)) = 1$$

If the term χ is not present in the document, $Presence(\mathcal{M}_\mathcal{D}, \chi) = w0$ with $w0$ is the null term.

More the term χ is present in the document $\mathcal{D}$, more the probability $P(t_{q,i} | \mathcal{M}_\mathcal{D})$ is high.

7.1.2. The term is absent in the document:

We use a smoothing function to take into account that a term of the query can be absent in the document and to avoid a null score for the entire query. There are lots of smoothing functions in language model domain; we choose to use a smoothing by interpolation proposed by Hiemstra [5]. The probability of the term $t_{q,i}$ in the corpus $\mathcal{C}$ is expressed by $P(t_{q,i} | \mathcal{M}_\mathcal{C})$.

$$P(t_{q,i} | \mathcal{M}_\mathcal{C}) = \frac{df(t_{q,i})}{\sum_{t_j \in \mathcal{VI}} df(t_j)} \quad \text{with} \quad df(t_{q,i}) \text{ the number of documents from } \mathcal{C} \text{ containing } t_{q,i} \text{ and } \mathcal{VI} \text{ the indexation vocabulary of } \mathcal{C}$$

7.1.3. Some approximations of the term are present in the document:

$Approximation(\mathcal{M}_\mathcal{D}, \chi)$ determines all the terms of the document $\mathcal{M}_\mathcal{D}$ can be paired with χ such as $\chi \neq y$.

$$Approximation : CI \times \mathcal{V} \rightarrow \mathcal{V}^*$$

$$(\mathcal{M}_\mathcal{D}, \chi) \rightarrow \left\{ t_j \in \mathcal{M}_\mathcal{D} \mid Absence(\mathcal{M}_\mathcal{D}, \chi) = 0 \text{ and } Presence(\mathcal{M}_\mathcal{D}, \chi) \neq t_j \right\}$$

$$P(\sim t_{q,i} | \mathcal{M}_{\mathcal{D}}) = \frac{\sum_{t_{d,j} \in \text{Approximation}(\mathcal{M}_{\mathcal{D}}, t_{q,i})} \text{ValConc}(t_{q,i}, t_{d,j}) \times \text{ValInter}(t_{q,i}, t_{d,j}) \times P(t_{d,j})}{|\text{Approximation}(\mathcal{M}_{\mathcal{D}}, t_{q,i})|}$$

To determine approximations of a term, we use the algorithm of Soundex [11].

7.2. Matching function

To combine the three dimensions: presence of the term in the document, presence of the term in the corpus and presence of the approximations of the term in the document, we must use two parameters. To define the importance of approximations in comparison with the terms themselves, the parameter λ is introduced. The parameter μ allows to vary the importance of the smoothing function (i.e. $x \neq y$).

It is also necessary to take into account the fact that terms extraction of the query are uncertain introducing $\text{Cert}(t_{q,i})$ in the matching function. So, the matching function taking into account uncertainty notion is:

$$P(Q|\mathcal{D}) = \prod_{i=1}^{\mathcal{N}_q} \text{Cert}(t_{q,i}) \times [\mu\lambda P(t_{q,i} | \mathcal{M}_{\mathcal{D}}) + \mu(1-\lambda)P(t_{q,i} | \mathcal{M}_{\mathcal{C}}) + (1-\mu)P(\sim t_{q,i} | \mathcal{M}_{\mathcal{D}})]$$

8. FIRST EVALUATIONS

To validate the integration of term certainty value in term weighting, we evaluated our proposition on CLEF-2004 collection. This corpus is composed of French newspapers articles ("Le Monde"). There are 47 646 documents and 50 queries in natural language. We compare our proposition [10] with Salton's vectorial model [8]. Our term weighting improves precision for the first recall value (cf. Tab 1).

Recall	Vectorial model	Our term weighting
0	0,4129	0,4555
0.1	0,3800	0,4066

Tab 1. Comparison between Salton's vectorial model and our model

9. CONCLUSION AND PERSPECTIVES

In this paper, we deal with the problem of uncertain data (for example data from ASR or OCR process). There is no information retrieval system in response to this problem: only some observations that retrieval performances are not significantly affected with the use of long documents and more affected with short documents. We want to take into account this dimension in the matching function. To make it, we introduce two notions: term certainty value and pairing between two terms. In a near future, we will evaluate our model. We ask the question to know if this method cannot be seen like a language model smoothing method. The handling of the uncertainty can be a solution for absent words in document.

REFERENCES

- [1] J. Allen, Maintaining knowledge about temporal intervals, Communications of the ACM, **26**(11), 832–843
- [2] W. Cooper, Expected search length: A single measure of retrieval effectiveness based on the weak ordering action of retrieval systems. American Documentation, 19(1), 30-41, 1968
- [3] W.B. Croft, S.M. Harding, K. Taghva and J. Borsack, An evaluation of information retrieval accuracy with simulated OCR output, in Proceedings of the third annual symposium on document analysis and information retrieval, pages 115-126, Las Veags, NV, April 1994
- [4] D. Grangier, A. Vinciarelli, H. Bourlard, Information retrieval on noisy text, Septembre 2003
- [5] D. Hiemstra, A linguistically motivated probabilistic model of information retrieval, dans Christos N and Stephanides C. (eds), Proc. European Conference of Digital Library (ECDL98), Sept. 1998, Springer Verlag
- [6] E. Mittendorf, P. Schäuble and P. Sheridan, Applying probabilistic term weighting to OCR text in the case of a large alphabetic library catalogue, SIGIR'95, pages 328-335, Seattle, 1995
- [7] J.M. Ponte and W.B. Croft, A language modelling approach to information retrieval, Research and Development in Information Retrieval, Proc. ACM-SIGIR, pp 275-281, 1998
- [8] Salton G (1971), The SMART Retrieval System - Experiment in Automatic Document Processing, Prentice-Hall, Englewood Cliffs, New Jersey
- [9] K. Taghva, Julie Borsack, Allen Condit and Srinivas Erva, The effects of noisy data on text retrieval, may 1993
- [10] C. Tambellini, C. Berrut, Pondération des données incertaines dans les systèmes de recherche d'informations : une première approche expérimentale, INFORSID 2006, Hammamet, Tunisie, 1-4 juin 2006
- [11] <http://en.wikipedia.org/wiki/Soundex>