Passage retrieval in context: Experiments on Patents.

Lucas Albarede[*,] — Philippe Mulhem[*] — Lorraine Goeuriot[*] Claude Le Pape-Gardeux[**] — Sylvain Marie[**] — Trinidad Chardin-Segui[**]**

[*] *Laboratoire d'Informatique de Grenoble, Email: firstname.lastname@imag.fr*
[**] *Schneider Electric Industries SAS, Email: firstname.lastname@se.com*

RÉSUMÉ.

ABSTRACT. *Focused retrieval retrieves and ranks sub-parts of documents according to their estimated relevance to a query. Many approaches akin to Structured Document retrieval exploit documents structure to effectively retrieve logic elements (titles, sections, etc...). Other approaches like Passage Retrieval aim at retrieving arbitrary length text unit (passages), considering the document as a unstructured flat text. In this work, we use the best of the two worlds. We want to* ***(1)*** *retrieve passages to find the best text units to retrieve ;* ***(2)*** *exploit the document's structure to more effectively estimate the passages' relevance. Previous work has shown that leveraging on the passage context was efficient for passage ranking. We believe that the information given by a document's structure can be used to estimate its passages' context. Firstly, we propose several ways to represent and integrate a document's structure and its sub-structures elements (sections) in the estimation of its passages' context. Secondly, we integrate these passage contexts in a state-of-the-art passage retrieval model. We evaluate our approach on two passage retrieval tasks on structured documents: CLEF IP2012 and CLEF IP2013. Our results show that using a document's structure to estimate its passages' contexts improves retrieval performances.*

MOTS-CLÉS : *Recherche d'information, Recherche de passage, Recherche de brevet, Contextualisation, Documents structurés.*

KEYWORDS: *Information Retrieval, Passage Retrieval, Patent Retrieval, Contextualization, Structured documents.*

1. Introduction

Structured retrieval (or XML retrieval) is concerned with the retrieval of document elements. The structure of a document, most of the time provided by the document mark-up language, is exploited to find the most relevant document elements to a query. One consequence is that the different retrieved elements have varying length, increasing or decreasing depending on their level in the hierarchy. Indeed, leaf elements often take the form of small text units. This approach brings two problems: **(1)** As the terms of a query are less likely to directly appear in a small text unit, such text units might be ranked lower despite their relevance. **(2)** The elements defined beforehand from the mark-up language bound the retrieval and prevent the system from returning potentially relevant text units that are not anchored in the document's structure.

To cope with problem **(1)**, current approaches resort to *contextualization* ; that is, the consideration of an element's context. Passage retrieval is concerned with the retrieval of passages: small textual elements. Its objective is very similar to Structured Retrieval, and share the same need for *contextualization*. However, unlike these, passage retrieval is purposed to work with unstructured documents. To do so, it usually segments a document into passages according to hand-crafted heuristics based on the number of characters, words, or punctuation. This allows for a free segmentation of a document's textual content, coping with problem **(2)**.

In this work, we investigate the combination of approaches akin to Structured Retrieval exploiting the structure of a document and approaches akin to Passage Retrieval allowing a document's textual content to be freely segmented. We argue that *contextualizing* a passage amounts to propagate relevance from other passages of the same document towards it. We explore several propagation methods. More precisely, we hypothesize that passages closer to each other might be able to better contextualize each other. Evaluation performed on two patent passage retrieval tasks (CLEF-IP2012, CLEF-IP2013) shows the merits of our approach.

We first present in section 2 a state of the art focusing on Passage Retrieval and Structured Retrieval. Then, in section 3, we dive into our proposals starting with some definitions and then investigating relevance propagation as a *contextualization* tool. Finally in section 4, we conduct and evaluation on two patent passage retrieval tasks. We present and then analyze the results before concluding.

2. State of the Art

Passage Retrieval

Several *contextualization* methods for passage retrieval were examined in the past. A commonly used context for passages is their containing document (Sheetrit *et al.*, 2019 ; Callan, 1994 ; Murdock et Croft, 2005 ; Fernández *et al.*, 2011 ; Bendersky et Kurland, 2008). We will use such idea in our proposal.

Others passage *contextualization* approaches consider its neighbour passages (Sheetrit *et al.*, 2019 ; Fernández *et al.*, 2011 ; Krikon *et al.*, 2011 ; Carmel *et al.*, 2013). For instance, (Sheetrit *et al.*, 2019) considers the previous and next passages of a passage. In our case, we consider a passage's neighbourhood according to the structure of its document.

Query terms proximity may be used to estimate a passage's context (Carmel *et al.*, 2013 ; Beigbeder, 2010). These approaches, based on Position Language Models (Lv et Zhai, 2009), give each position in a text a proximity value, depending on its distance from the query terms : each query term propagates its relevance in a decreasing manner around it. We also share the intuition that a relevant element can positively influence its neighbourhood decreasingly in term of distance. We investigate in our work the propagation of relevance not between terms, but between passages, and more generally elements of a document's structure.

Structured Retrieval

Structured retrieval methods represent a document's structure as a tree. Their objective is to leverage on this tree to find relations between its elements to perform *contextualization* (Kekäläinen *et al.*, 2018).

Some approaches (Norozi et Arvola, 2013 ; Norozi *et al.*, 2012 ; Arvola *et al.*, 2005 ; Arvola *et al.*, 2008) introduce the notion of neighbourhood *contextualization*. They propose to contextualize elements in their "neighbourhood". They propagate an element's relevance in an uniform manner across its neighbourhood. We expand such works by proposing weighted relevance propagation depending on its distance from the propagating element.

Other approaches close to our work are (Callan, 1994 ; Kaszkiel *et al.*, 1999 ; Ogilvie et Callan, 2005 ; Mass et Mandelbrod, 2005). They argue that the relevance score of a non-leaf node should depend of its children scores. Furthermore, they use these new founded scores to smooth down the relevance of their children.

Our approach use comparable mechanisms, under the form of relevance propagation, but differ in two points. First, these approaches use heavy marked up documents (such as XML documents) which contain lots of different tags (entry, weblink, link, b, lists, ref...) segmenting a document into very small elements (Norozi *et al.*, 2012 ; Norozi et Arvola, 2013). We position ourselves in a situation closer to Passage Retrieval and consider a document's structure to be composed solely of *sections* (non-leaves nodes) and *passages* (leaves nodes) which are extracted from the a flat representation of a section's text content.Moreover, we consider only the retrieval of passages and do not bother with the retrieval of other structuring elements. Second, we consider relevance propagation such propagation should be weighted by a decreasing function of the structural distance between the propagating element and its target.

3. Proposal

3.1. *Definitions*

A structured document is composed of a logical structure and a textual content (Verbyst, 2008 ; Lalmas, 2009). A document's structure is usually represented as a tree in Structured Retrieval (Ogilvie et Callan, 2005 ; Norozi et Arvola, 2013 ; Norozi *et al.*, 2012). However, we consider here the general structured document case and do not consider any particularity about mark-up tags : we consider the structure to be solely composed of *sections* and *passages*. An example of how we represent a document's structure can be seen in figure 1.

A *section* is a titled recursive logical element that may contain other non-leaves nodes *sections* with titles, or leaves nodes *passages* without title.

A *passage* is a textual element that has a single parent section. Passages may be defined by an approximate length (number of characters or words) or delimited by punctuation. In all cases, a passage is bound by the section which contains it.

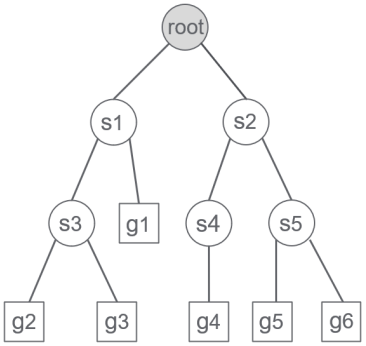

Figure 1. *Representation of a document's structure. Circles (**root, s1, s2, s3, s4, s5**) and squares (**g1, g2, g3, g4, g5, g6**) represent sections and passages, respectively.*

The following formal notations and expressions will be used throughout the paper.

A document *d* is characterized by its root section $root_d$. It also possesses a set of passages and sections.

A section *s* is characterized by the set of sections s_s and passages s_g that are its direct children. Moreover, s_{title} indicates *s*'s title.

A passage[1] *g* is characterized by its document (the document in which it appears in) g_{doc}, and its direct parent section $g_{section}$. As a matter of clarity, we note that *g* belong to several sections: one directly and several transitively.

1. Notations taken from the literature (Sheetrit *et al.*, 2019 ; Sheetrit et Kurland, 2019)

3.1.1. *Retrieval Framework*

The purpose of passage retrieval is to output a ranked list of passages, giving each of them a relevance score with respect to a query. In this paper, we denote the scoring function by $G_{retrieval}$, and a passage relevance score by **final passage score**. This is done to avoid any confusion since our models might use intermediary passage scores in the computation of the **final passage score**.

As already stated, previous works showed that contextualizing passages is important to correctly estimate their relevance to a query. We argue that $G_{retrieval}$ should compute a passage's **final passage score** using information about its textual content and information about its context.

$$G_{retrieval}(q, g) = information\_content(q, g) \oplus information\_context(q, g) \quad [1]$$

3.2. *Relevance Propagation as a Contextualization Tool*

Many state-of-the-art methods contextualize a passage *g* by combining and mixing scores from other passages of the document or even use the whole document itself (Carmel *et al.*, 2013 ; Beigbeder, 2010 ; Sheetrit *et al.*, 2019 ; Callan, 1994 ; Murdock et Croft, 2005 ; Fernández *et al.*, 2011 ; Bendersky et Kurland, 2008). This amounts to "propagate" relevance information between the different elements of a document.

We consider various methods to propagate relevance information through the structure of a document. We have the intuition that closer elements are more likely to contextualize each other. Behind this intuition we hypothesize that, to improve *contextualization*, a relevance propagation should be relative to the distance (in the tree) between two elements.

First, we present our method used to measure the similarity between a text and the query, used in the remainder of this paper. Then, we will look into absolute relevance propagation methods where a propagated relevance is equivalent for every element targeted by the propagation as seen on Figure 2 (a). Finally, we investigate relative relevance propagation methods where a propagated relevance depends on the distance between the propagator and the element targeted by the propagation as seen on Figure 2 (b).

3.2.1. *Text-Query Similarity*

In the following, all initial matching between a text item (i.e: document, section title, passages, ...) are based on the negative Cross Entropy (Zhai et Lafferty, 2001 ; Sheetrit *et al.*, 2019 ; Sheetrit et Kurland, 2019) between the unigram language models induced from them:

$$Sim(x, y) = exp(-CE(\theta_x^{MLE} || \theta_y^{Dir})) \quad [2]$$

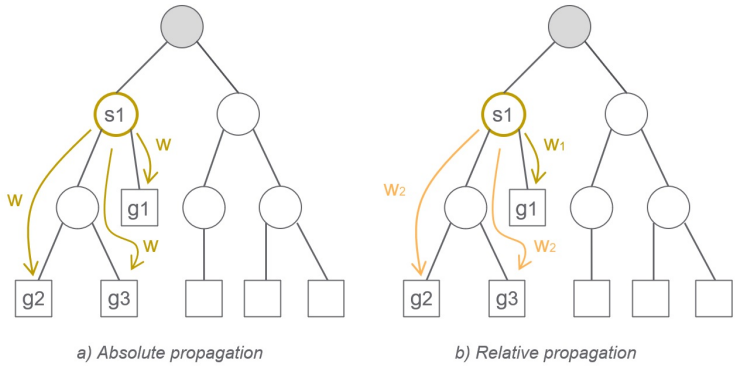

Figure 2. *Differences between absolute propagation (a) and relative propagation (b).* ***s1*** *is a section propagating relevance to its passages* ***g1, g2, g3*** *uniformly on the left, and depending on their relative distance on the right.*

θ_x^{MLE} is the unsmoothed maximum likelihood estimate induced from x and θ_y^{Dir} is a Dirichlet smoothed language model induced from y (Zhai et Lafferty, 2001). This similarity function has shown good performances for retrieval applications.

3.2.2. ***Absolute Relevance Propagation***

paragraph****Document-Only Contextualization***

Our first proposal, **QSF**$_v$, is a variant of the "query-similarity fusion" function (Callan, 1994 ; Carmel *et al.*, 2013). The **final passage score** combines an initial (out of context) score from the passage and a score from its document.This linear combination propagates the relevance from a document's score to its passages: each passage of the same document gets the same propagation of relevance.

The initial (out of context) score of a passage estimates the $information\_content(q, g)$ part of equation (1), while the propagated document's score estimates its $information\_context(q, g)$.

The process is as follow: we initially compute an initial score (with respect to a query q) for every passage and every documents using *Sim(x,y)*. Then, we apply a min-max normalization to the passage scores on one hand and the document scores on the other hand. Such normalization is a must, as the full document is much larger than a passage, leading to large differences in the matching scores. Finally, we fuse with a linear combination the passage normalized score and its document g_{doc} *normalized score*. We use min-max normalization because it brings both scores to [0,1] scale, leading to better explainability of the linear combination. We note that linear combination is chosen for simplicity and leave to future works the analysis of more precautionary forms of combinations such as the one presented in (Robertson *et al.*, 2004).

We define the function **QSF**$_v$ as:

$$QSF_v(q,g) = \alpha * Sim_{norm}(q,g) + (1-\alpha) * Sim_{norm}(q, g_{doc}) \quad [3]$$

α being a free parameter : $\alpha \in [0,1]$ Where $Sim_{norm}(q,g)$ is computed as such :

$$Sim_{norm}(q,g) = \frac{Sim(q,g) - \min_{\{g' \in G_{init}\}} Sim(q,g')}{\max_{\{g' \in G_{init}\}} Sim(q,g') - \min_{\{g' \in G_{init}\}} Sim(q,g')} \quad [4]$$

$Sim_{norm}(q, g_{doc})$ is computed the same way, replacing $Sim(q,g)$ with $Sim(q, g_{doc})$.

Titles-Only Propagation

The different elements contained in a section (be it passages or other sections) are usually semantically linked. We argue that this link can be characterized by the sections' titles, and think that we can use them to contextualize their passages. For example, the words in a section's title does not always appear in the passages contained in said section, meaning that these passages are missing some *contextualization* information. If a passage is missing some of the query words, but these words appear in its parent sections' titles, the passage score will be increased.

To cope with this behaviour, each passage's relevance is impacted by the relevance of its parent sections' titles: we propagate the relevance information about a section's title to its passages.

A passage may be, directly of transitively, part of several sections. However it is difficult to know if the title of its direct parent section is of more importance than the one of the document, for example. Because of that, we decide to propagate titles information from a section to its passages equivalently.

Since titles are usually very small text units (a few words) we argue they would not fit as a proper "document" for language model based approaches (such as our similarity $Sim(x,y)$) to work. That is why we choose to modify the way we initially score a passage using the titles of the sections it belong to. We design $Sim_{title}(q,g)$ as a query similarity measure that takes into account the words in passage g and the words in every of g's parent section, from its direct parent section $g_{section}$ to the root of the document $root_d$:

$$Sim_{title}(q,g) = Sim(q, g \oplus_{\forall s : g_{section} \rightarrow root_{g_{doc}}} s_{title}) \quad [5]$$

where the operator $\oplus$ indicates a concatenation between two texts, and the operator $\rightarrow$ indicates an enumeration of the sections from $g_{section}$ to $root_{g_{doc}}$.

For the remainder of this paper, we will use $Sim_{title}(q,g)$ instead of $Sim(q,g)$ when initially scoring a passage g and define the function $QSF_{v_{title}}(q,g)$ which is a variant of $QSF_v(q,g)$ using this new similarity function:

$$QSF_{v_{title}}(q,g) = \alpha * Sim_{title_{norm}}(q,g) + (1-\alpha) * Sim_{norm}(q, g_{doc}) \quad [6]$$

α being a free parameter : $\alpha \in [0,1]$

Direct Parent Section Contextualization

As stated before, a section links the elements it contains in a semantic way, similarly to a full document. However, a section is a more focused semantic element than a document: two passages *a* and *b* appearing in the same section have more chances to be semantically related than two passages *c* and *d* appearing in different sections but in the same document. We also think that, out of all the parent sections of a passage, its direct parent section holds the most precise semantic information. Thus, we argue that considering a passage's direct parent section can effectively contextualize it. We compute a relevance score for every passages' direct parent section, and integrate this score into equation [1] by uniformly propagating a section's score to its direct children.Similar to (Callan, 1994 ; Kaszkiel *et al.*, 1999 ; Ogilvie et Callan, 2005), we define the score given to a section *s* as an aggregation of its direct children's score. The initial score of a passage is $Sim_{title}(q, g)$ (as defined by equation [5]). The aggregation function might be any non-decreasing function, such as, for example, average or maximum.

Formally we recursively define the function $Sim_{sec}(q, s)$ as:

$$Sim_{sec}(q, s) = aggregation(\left[Sim_{title}(q, g)\right]_{\forall g \in s_g}; \left[Sim_{sec}(q, s')\right]_{\forall s' \in s_s}) \quad [7]$$

To be compliant with the definition of $G_{retrieval}$, $Sim_{sec}(q, s)$ should estimate the $information\_context(q, g)$ component. We first decide to combine it with the query-document similarity ($Sim_{norm}(q, g_{doc})$) since the two components bring context information from different granularity levels, so as to improve the context estimation. This amounts to modify $\mathbf{QSF}_v$ by integrating $Sim_{sec}(q, s)$ in its context estimation. To do so, we first follow the same normalization process as before: We define $Sim_{sec_{norm}}(q, g_{section})$ as the min-max normalized score of passage *g*'s direct parent. Then, we compute *g*'s **final passage score** by integrating a linear combination between this normalized score and the normalized score of *g*'s document. Formally, we define $\mathbf{QSF}_{section}$ as:

$$\begin{aligned} QSF_{section}(q, g) =& \alpha * Sim_{title_{norm}}(q, g) \\ &+(1-\alpha) * \Big(\beta * Sim_{norm}(q, g_{doc}) \\ &+(1-\beta) * Sim_{sec_{norm}}(q, g_{section}) \Big) \end{aligned} \quad [8]$$

α and β being free parameters : $\alpha, \beta \in [0, 1]^2$.

3.2.3. ***Relative Relevance Propagation***

We investigate here "distance-based" relevance propagation through a document's structure. We define such distance between 2 document elements x and y, *distance(x,y)*, as the number of edges on the graph required to move from one element to the other.

Full Parent Section Contextualization

We have previously derived a function $\mathbf{QSF}_{section}$ that contextualizes a passage with its direct parent section. However, a passage often has multiple parent sections and it is natural to investigate their ability to contextualize efficiently. This method has already shown its strength (Arvola *et al.*, 2008). Here we will in addition exploit the relative distance between elements.

We think that it is possible to effectively contextualize a passage g by propagating its parent sections (i.e., all the sections that contain, directly of transitively, the passage) scores according to their relative distance to g with the following intuition : the farther from a passage a parent section is, the less its score should be propagated. To do so, we define $dist_{weight}(distance(g,s))$ as a strictly decreasing function of *distance(g,s)* (the distance between g and s on the graph). Our goal is to define $propag_{section}(q,g)$ as a function aggregating g's parent sections similarity scores weighted by $dist_{weight}$, and integrate it into $G_{retrieval}$. An example of weighted relevance propagation from parent sections can be seen in figure 3.

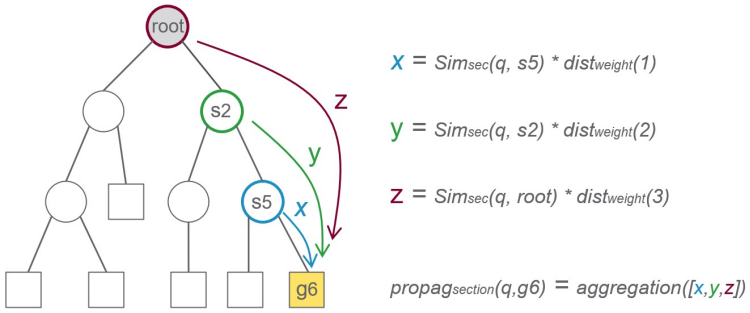

Figure 3. *Example of relevance propagation from parent sections. g6 represents the passage targeted by the propagation. root, s2 and s5 are sections. q is a query.*

This process is done in four steps : **(1)** we score each section in a document d with equation (7), **(2)** for each passage in d we compute propagation scores from its different parent sections. This results in a set of propagated scores for each passage. Then **(3)** we aggregate these propagated scores and give each passage a single $propag_{section}$. Finally **(4)** we modify $\mathbf{QSF}_v$ by integrating these scores, creating $\mathbf{QSF}_{sectionPropagate}$.

More formally, we define the function *propag* as :

$$propag_{section}(q,g) = aggregation\Big(\big[Sim_{sec}(q,s) * dist_{weight}(distance(g,s))\big]_{s:g_{section}\rightarrow root_{g_{doc}}}\Big) \quad [9]$$

where $g_{section} \rightarrow root_{g_{doc}}$ indicates the enumeration of parent sections from $g_{section}$ to $root_{g_{doc}}$.

We follow the same principle as we did with $\mathbf{QSF}_{section}$ by considering that the $propag_{section}$ scores should be integrated into the $information\_context(q, g)$ component of $G_{retrieval}$. We start with a min-max normalization of the $propag_{section}$ scores into $propag_{section_{norm}}$ scores and compute g's **final passage score** by integrating them into the $\mathbf{QSF}_v$:

$$\begin{aligned} QSF_{sectionPropagate}(q,g) =& \alpha * Sim_{title_{norm}}(q,g) \\ &+(1-\alpha) * \Big(\beta * Sim_{norm}(q, g_{doc}) \\ &+(1-\beta) * propag_{section_{norm}}(q, g_{section}) \Big) \end{aligned} \quad [10]$$

α and β being free parameters : $\alpha, \beta \in [0,1]^2$.

Passage Relevance Propagation

We now investigate a more direct approach to weighted relevance propagation. We explore if we can directly contextualize a passage by propagating the relevance of all other passages in the same document.

The process is very similar to the one in $QSF_{sectionPropagate}(q, g)$, except that do not propagate from parent sections, but directly from every passage of a document.

First, we define $propag_{passage}$ as :

$$\begin{aligned} propag_{passage}(q,g) =& aggregation\Big(\big[Sim_{title}(q, g') \\ &* dist_{weight}(distance(g,s)) \big]_{g' \in g_{doc}, g' \neq g} \Big) \end{aligned} \quad [11]$$

Then, we min-max normalize the $propag_{passage}$ scores into $propag_{passage_{norm}}$ scores and compute g's **final passage score** by integrating them into $\mathbf{QSF}_v$ as such :

$$\begin{aligned} QSF_{passagePropagate}(q,g) =& \alpha * Sim_{title_{norm}}(q,g) \\ &+(1-\alpha) * \Big(\beta * Sim_{norm}(q, g_{doc}) \\ &+(1-\beta) * propag_{passage_{norm}}(q, g) \Big) \end{aligned} \quad [12]$$

α and β being free parameters : $\alpha, \beta \in [0,1]^2$

4. Evaluation

4.1. *Experimental Setup on CLEF-IP Tasks*

Datasets

Since our approach exploits the structure of documents, we focus our evaluation on highly structured documents: patents. We evaluate our models on the CLEF-IP2012 and CLEF-IP2013 passage retrieval tasks (Piroi *et al.*, 2012 ; Piroi *et al.*, 2013). These tasks are both based on the CLEF-IP dataset, which contains 2.6 million patent documents, and contains French, German and English queries separated in train and test sets. We conduct our experiments on English queries only. For CLEF-IP2012 this amounts to 21 training queries and 35 test queries. For CLEF-IP2013 this amounts to 56 training queries and 50 test queries.

Structure Extraction

Patents in the CLEF-IP dataset are available in XML format. They are most of the time segmented in four main sections: bibliography, abstract, description and claims. To build the structure of a patent document, we use these four sections as starting points in the XML structure to look for other sections. We use hand-crafted features, either based on XML tags, case or number of characters.We segment a patent document into passages which align with the relevance judgements of both tasks.

Query Transformation

The objective of the two aforementioned tasks is prior art search: finding patents (in this case passages of patents) that are similar to a set of query claims coming from a patent document. It is a popular practice to use this full patent document and transform it into a short, refined query (Mahdabi *et al.*, 2011 ; Xue et Croft, 2009 ; Mahdabi *et al.*, 2013 ; Andersson *et al.*, 2016). We use an already solid method from the state-of-the-art (Mahdabi *et al.*, 2011). Let q_d be a query patent document, our implementations defines a first form of the query as the 10 words with highest *tf-idf* in q_d's abstract, and a second form of as the 100 words with highest *tf-idf* in q_d's query claims.

Evaluation Measures

CLEF-IP2012-2013 results were reported according to five evaluation measures. We report the same measures to be compliant with the original tasks. Here, a relevant document is a document which contains at least one relevant passage. We report three measures at the document-level: **(1)** PRES@100 which measures the effectiveness of ranking documentsrelative to the best and worst ranking cases, where the best ranking case is retrieving all relevant documents at the top of the list, and the worst is retrieving all the relevant documents just after the maximum number of documents to be checked by the user[2] (in this case, 100) (Magdy et Jones, 2010). **(2)** RECALL@100 and

2. http://homepages.inf.ed.ac.uk/wmagdy/PRES.html

(3) MAP@100. We also report two measures at the passage level: **(4)** MAP(D) which computes the AP inside each relevant document (considering its passages), averages this score for a query over its relevant documents, and averages it across all queries to get the MAP. **(5)** PREC(D) which computes the precision inside each relevant document and averages the scores in the same manner as for MAP(D).

4.1.1. *Models Implementation*

Efficient Adaptation of the Framework

As it is usual for passage retrieval, our models process a query with a document retrieval step and then a passage ranking step. A set of documents is first retrieved using the query,and then the model computes scores for every passage in this document set. During the document retrieval step, we use a filtering mechanism to eliminate documents which do not share any IPC code (International Patent Classification: codes grouping patents according to different criteria) with the query patent document. Even if this filtering might discard relevant documents, it brings good performances in practice (Gobeill et Ruch, 2012).

We set the number of documents to be retrieved (during the first step) to 1000. We also set the number of passages to be returned by the system to 1500, which means that the second step ranks every passage of the 1000 documents, and keeps the 1500 highest-ranked ones.

We implement our models using the Terrier Information Retrieval System (Ounis *et al.*, 2006). We use a classical porter stemmer and discard stop-words according to the stop-words list native to Terrier. The Dirichlet smoothing parameter inside $Sim(x, y)$ and $Sim_{title}(x, y)$ is set to 1000 (Sheetrit *et al.*, 2019 ; Zhai et Lafferty, 2001 ; Sheetrit et Kurland, 2019), and we learn every other parameter by optimising the MAP(D) measure on the train set.

Table 1 summarizes the values we learn the different parameters on.

Aggregation Function

Three of our models use an *aggregation* function to compute the score of a passage. For $\textbf{QSF}_{section}$, we choose the *aggregation* function to be the *average* function following the literature (Ogilvie et Callan, 2005 ; Norozi et Arvola, 2013 ; Arvola *et al.*, 2005). For $\textbf{QSF}_{sectionPropagate}$ and $\textbf{QSF}_{passagePropagate}$, we experimented with two implementations of this function : *average* and *maximum*. However we found that the *maximum* function leads to significantly worse performances. We choose to only report the results of the *average* implementations under the names $\textbf{QSF}_{sectionPropagateAVG}$ and $\textbf{QSF}_{passagePropagateAVG}$.

Distance Weighting Function

Two of our models use the distance between two elements $dist_{weight}(distance(x, y))$ in the document structure to compute the score of a

	α	β	Gaussian parameter	Dirichlet parameter
Range	$0 \rightarrow 1$	$0 \rightarrow 1$	[0.5, 1, 2, 5]	[1000]
Step	0.1	0.1	-	-

Tableau 1. *Range and step of values the model parameters are learned on.*

passage. We have defined *distance(x,y)* in Section 3, and deliberately left $dist_{weight}$ subject to experiments. Following works using propagation functions (Carmel *et al.*, 2013), we define $dist_{weight}(distance(x, y))$ as the Gaussian function:

$$dist_{weight}(distance(x, y)) = e^{-\frac{distance(x,y)^2}{2\sigma^2}} \quad [13]$$

σ being a free parameter. We experimented with values of σ in {0.5, 1, 2, 5}.

4.2. *Results*

4.2.1. *Models Performances*

Tables 2 and 3 report the performance of several passage retrieval models for the passage retrieval task of CLEF-IP2012 and CLEF-IP2013, respectively. The first rows report the best scores obtained at CLEF-IP2012-2013 as measured by the five evaluation measures: they do not necessarily come from the same run[3][4]. On Table 1, from the second row up to the end of the table, we present our models and their variations.

On Table 2, we also present a more recent approach which focuses on query generation (Andersson *et al.*, 2016). We note that this work is dissimilar to us because it focuses on transforming a patent query into a clever, more refined query by applying several filters to candidate words before selecting them .Though, we report their results to be closer to the state of the art.Out of all the approaches they present in the paper, we chose to report the one which had the highest MAP(D) and PREC(D).

We can see on Table 2 that our approaches outperform the best results reported on CLEF-IP2012 for four out of the five evaluation measures. We note on Table 3 that we outperform the best results reported on CLEF-IP2013 and those reported by a more recent work for three out of the five evaluation measures.

The increases in PRES@100 and Recall@100 across both tasks indicate that our *contextualization* approaches, especially $QSF_{passagePropagateAVG}$ and $QSF_{sectionPropagateAVG}$, are able to retrieve passages for a greater number of different relevant documents.

3. CLEF-IP2012 references for PRES, Recall, MAP, MAP(D) are (Gobeill et Ruch, 2012). Reference for PREC(D) is (Wilhelm-Stein *et al.*, 2012)

4. CLEF-IP2013 references for PRES, Recall, MAP, PREC(D) are (Luo et Yang, 2013). Reference for MAP(D) is (Eiselt et Oberreuter, 2013)

Methods	PRES@100	Recall@100	MAP@100	MAP(D)	PREC(D)
IP-2012 top scores	0.3313	0.4401	**0.1410**^{ij}	0.0964	0.1032
QSF_v	0.3545^{o}	0.4901^{o}	0.1071	0.1392^{o}	0.1150
$QSF_{v_{title}}$	0.3570^{o}	0.5152^{o}	0.1110	0.1562^{o}	0.1310^{o}
$QSF_{section}$	0.3578^{o}	0.5307^{oi}	0.1372^{i}	0.1643^{o}	0.1249
$QSF_{sectionPropagateAVG}$	0.3630^{oi}	0.5401^{oij}	0.1379^{i}	0.1755^{oij}	**0.1328**^{o}
$QSF_{passagePropagateAVG}$	**0.3700**^{oijk}	**0.5474**^{oij}	0.1382^{i}	**0.1808**^{oij}	0.1301^{o}

Tableau 2. *Performance over CLEF-IP2012 (in boldface: best result in a column). o, i, j, k, l and m represent statistical significance (two tailed paired t-test, p≤5%) over the top IP-2012 score, QSF_v, $QSF_{v_{title}}$, $QSF_{section}$, $QSF_{sectionPropagateAVG}$ and $QSF_{passagePropagateAVG}$ respectively.*

Methods	PRES@100	Recall@100	MAP@100	MAP(D)	PREC(D)
IP-2013 top scores	0.4327	0.5399	**0.1912**^{ijkm}	0.1416	0.2140^{ij}
(Andersson *et al.*, 2016)	0.444	0.560	0.187	0.146	**0.282**
QSF_v	0.4383	0.5705^{o}	0.1581	0.1631^{o}	0.1870
$QSF_{v_{title}}$	0.4412	0.5764^{o}	0.1645	0.1800^{o}	0.1990
$QSF_{section}$	0.4522^{o}	0.5810^{o}	0.1682	0.1878^{oi}	0.2189
$QSF_{sectionPropagateAVG}$	**0.4613**^{oij}	**0.6034**^{oij}	0.1731^{i}	**0.2073**^{oijh}	0.2391^{oij}
$QSF_{passagePropagateAVG}$	0.4568^{oi}	0.5917^{oi}	0.1699^{i}	0.1978^{oij}	0.2303^{oi}

Tableau 3. *Performance over CLEF-IP2013 (in boldface: best result in a column). o, i, j, k, l and m represent statistical significance (two tailed paired t-test, p≤5%) over the top IP-2013 score, QSF_v, $QSF_{v_{title}}$, $QSF_{section}$, $QSF_{sectionPropagateAVG}$ and $QSF_{passagePropagateAVG}$ respectively. Statistical tests over (Andersson* et al.*, 2016) do not appear since we don't have access to the original runs.*

Nevertheless, our lower performances with MAP@100 show that we also retrieve passages from a greater number of non-relevant documents.

The statistical significant increases in MAP(D) for both CLEF-IP2012 and CLEF-IP2013 indicate that in each relevant document, we manage to better rank the passages according to their relevance.

Besides, we report worse PREC(D) results than (Andersson *et al.*, 2016). This indicates that even though we are able to better rank the retrieved passages in their respective documents, we still retrieve non-relevant passages contained in those documents.

This effect could be linked to our optimization criteria. Indeed, by optimizing the MAP(D) measure, it is possible that our *contextualization* methods learn to properly propagate relevance to passages would should be contextualized (high MAP(D)), but

Methods	α	β
QSF_v	0.8	-
$QSF_{v_{title}}$	0.9	-
$QSF_{section}$	0.6	0.1
$QSF_{sectionPropagateAVG}$	0.6	0.3
$QSF_{passagePropagateAVG}$	0.5	0.2

Tableau 4. *Optimal α and β in our proposals, according to MAP(D) on IP-2012 training set.*

as a side effect also propagate too much relevance to passages who should not be contextualized (low PREC(D)).

As a matter of comparison between our approaches, we see that $QSF_{v_{title}}$ outperforms QSF_v across all evaluation measures, implying that considering the titles of a passage's parent sections for context estimation leads to better performances. Moreover, we see that $QSF_{section}$ outperforms both $QSF_{v_{title}}$ and QSF_v, indicating that considering a passage's direct parent section is a better context estimator than only considering its document.

We see that the relative relevance propagation methods ($QSF_{sectionPropagateAVG}$ and $QSF_{passagePropagateAVG}$), both outperform our first three methods across almost every evaluation measure. This indicates that propagating relevance according to the distance between two elements is better for contextualizing a passage, or in other words, that two elements close to each other in the document's structure have a better chance of contextualizing each other.

It is though unclear which of these two approaches is better for *contextualization*, since they both (slightly) outperform each other on the two tasks. We think that these approaches, built upon the same intuition, achieve the same objective and only differ in their implementations.

4.2.2. ***Parameters Analysis***

In this section we analyze the importance of the components in $G_{retrieval}$, according to our different model implementations. Table 3 reports the value of the parameters α and β of formulas [3,8,10,12] for the best run of our models on CLEF-IP2012.

A high α value indicates that the $information\_content(q, g)$ component is more important than the $information\_context(q, g)$ component to estimate a passage's **final passage score** and vice-versa. A high β value indicates that the $Sim(q, g_{doc})$ is more important than the other context component (different with respects to models)to estimate a passage's **final passage score** and vice-versa.

For the QSF_v and $QSF_{v_{title}}$ approaches, the parameter α is very high, indicating that the estimated context has a minimal impact on the computation of a passage's **final passage score**. However, we can see that for the three other approaches, α is

has a lower impact ($\alpha \in [0.5, 0.6]$). This indicates that these approaches consider the context as important to compute the **final passage score**. We deduce that this is because they estimate the context of a passage more effectively.

Moreover we can see that the optimal β parameter is quite low ($\beta \leq 0.3$), indicating that the other context component (different with respects to models) plays a more important role in the estimation of a passage's context than the naive query similarity of its document.

Finally, we can see that the β value of the absolute relevance method ($QSF_{section}$) is lower than the ones of the relative methods ($QSF_{sectionPropagateAVG}$, $QSF_{passagePropagateAVG}$). This indicates that even though these latter methods have a better context estimation, there are still some passages that benefit from having their relevance propagated in an uniform way. For example, if a passage is very far from another one but can contextualize it well. This implies that there are still ways to improve the detection of contextualizing passages for accurate relevance propagation.

5. Conclusion

In this work, we investigated the combination of approaches akin to Passage Retrieval and Structured Retrieval and tried to leverage benefits from both worlds to perform passage retrieval on structured document. More precisely we tackle a problem inherent to retrieving small textual elements : *contextualization*.

Contextualization has been analyzed in both Passage and Structured Retrieval in the form of relevance propagation. We look into several mechanisms of relevance propagation across a document's structure and integrate them into a standard passage retrieval environment. Moreover, we hypothesize that passages closer to each other might be able to better contextualize each other.

Evaluation performed on two passage retrieval tasks (CLEF-IP2012, CLEF-IP2013) show the merits of using a document's structure to perform passage *contextualization*. Furthermore, we found that propagating a passage's relevance to another passage depending on the distance between them leads to better *contextualization*.

For future works, we would like to explore other distance weighting functions (other than the Gaussian function). It would also be interesting to analyze how our approaches can be used in harmony with methods necessitating a first retrieval step such as relevance feedback or learning-to-rank. Another investigation would be to adapt our approaches to cases where the MAP(D) measure is not a priority, such as high-recall systems, or systems requiring the least number of non-relevant results retrieved. Finally, we plan to extend our experiments on two datasets: *PatentMatch* (Risch *et al.*, 2020), another patent retrieval dataset, and the *INEX* dataset (Geva *et al.*, 2010) which is composed of Wikipedia documents.

Remerciements

This work has been partially supported by MIAI@Grenoble Alpes (ANR-19-P3IA-0003), as well as the Association Nationale de la Recherche et de la Technologie (ANRT).

6. Bibliographie

Andersson L., Lupu M., Palotti J. a., Hanbury A., Rauber A., « When is the Time Ripe for Natural Language Processing for Patent Passage Retrieval ? », *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, Association for Computing Machinery, New York, NY, USA, p. 1453–1462, 2016.

Arvola P., Junkkari M., Kekäläinen J., « Generalized Contextualization Method for XML Information Retrieval », *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, Association for Computing Machinery, New York, NY, USA, p. 20–27, 2005.

Arvola P., Kekäläinen J., Junkkari M., « The Effect of Contextualization at Different Granularity Levels in Content-Oriented Xml Retrieval », *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, Association for Computing Machinery, New York, NY, USA, p. 1491–1492, 2008.

Beigbeder M., « Focused Retrieval with Proximity Scoring », *Proceedings of the 2010 ACM Symposium on Applied Computing*, SAC '10, Association for Computing Machinery, New York, NY, USA, p. 1755–1759, 2010.

Bendersky M., Kurland O., « Utilizing Passage-Based Language Models for Document Retrieval », *Proceedings of the IR Research, 30th European Conference on Advances in Information Retrieval*, ECIR'08, Springer-Verlag, Berlin, Heidelberg, p. 162–174, 2008.

Callan J. P., « Passage-Level Evidence in Document Retrieval », *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, Springer-Verlag, Berlin, Heidelberg, p. 302–310, 1994.

Carmel D., Shtok A., Kurland O., « Position-Based Contextualization for Passage Retrieval », *Proceedings of the 22nd ACM International Conference on Information amp ; Knowledge Management*, CIKM '13, Association for Computing Machinery, New York, NY, USA, p. 1241–1244, 2013.

Eiselt A., Oberreuter G., « Innovandio S.A. at CLEF-IP 2013 », *in* P. Forner, R. Navigli, D. Tufis, N. Ferro (eds), *Working Notes for CLEF 2013 Conference , Valencia, Spain, September 23-26, 2013*, vol. 1179 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2013.

Fernández R., Losada D., Azzopardi L., « Extending the language modeling framework for sentence retrieval to include local context », *Inf. Retr.*, vol. 14, p. 355-389, 08, 2011.

Geva S., Kamps J., Lethonen M., Schenkel R., Thom J. A., Trotman A., « Overview of the INEX 2009 Ad Hoc Track », *in* S. Geva, J. Kamps, A. Trotman (eds), *Focused Retrieval and Evaluation*, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 4-25, 2010.

Gobeill J., Ruch P., « BiTeM Site Report for the Claims to Passage Task in CLEF-IP 2012 », *in* P. Forner, J. Karlgren, C. Womser-Hacker (eds), *CLEF 2012 Evaluation Labs and Workshop, Online Working Notes, Rome, Italy, September 17-20, 2012*, vol. 1178 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2012.

Kaszkiel M., Zobel J., Sacks-Davis R., « Efficient Passage Ranking for Document Databases », *ACM Trans. Inf. Syst.*, vol. 17, n° 4, p. 406–439, October, 1999.

Kekäläinen J., Arvola P., Junkkari M., *Contextualization in Structured Text Retrieval*, Springer New York, New York, NY, p. 611-613, 2018.

Krikon E., Kurland O., Bendersky M., « Utilizing Inter-Passage and Inter-Document Similarities for Reranking Search Results », *ACM Trans. Inf. Syst.*, December, 2011.

Lalmas M., *XML retrieval*, vol. 1, 01, 2009.

Luo J., Yang H., « Query formulation for prior art search - Georgetown University at CLEF-IP 2013 », 01, 2013.

Lv Y., Zhai C., « Positional Language Models for Information Retrieval », *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, Association for Computing Machinery, New York, NY, USA, p. 299–306, 2009.

Magdy W., Jones G. J., « PRES: A Score Metric for Evaluating Recall-Oriented Information Retrieval Applications », *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, Association for Computing Machinery, New York, NY, USA, p. 611–618, 2010.

Mahdabi P., Gerani S., Huang J. X., Crestani F., « Leveraging Conceptual Lexicon: Query Disambiguation Using Proximity Information for Patent Retrieval », *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, Association for Computing Machinery, New York, NY, USA, p. 113–122, 2013.

Mahdabi P., Keikha M., Gerani S., Landoni M., Crestani F., « Building Queries for Prior-Art Search », *in* A. Hanbury, A. Rauber, A. P. de Vries (eds), *Multidisciplinary Information Retrieval*, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 3-15, 2011.

Mass Y., Mandelbrod M., « Component Ranking and Automatic Query Refinement for XML Retrieval », *in* N. Fuhr, M. Lalmas, S. Malik (eds), *Advances in XML Information Retrieval*, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 73-84, 2005.

Murdock V., Croft W. B., « A Translation Model for Sentence Retrieval », *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Vancouver, British Columbia, Canada, p. 684-691, October, 2005.

Norozi M. A., Arvola P., « Kinship Contextualization: Utilizing the Preceding and Following Structural Elements », *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, Association for Computing Machinery, New York, NY, USA, p. 837–840, 2013.

Norozi M. A., Arvola P., de Vries A. P., « Contextualization Using Hyperlinks and Internal Hierarchical Structure of Wikipedia Documents », *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, Association for Computing Machinery, New York, NY, USA, p. 734–743, 2012.

Ogilvie P., Callan J., « Hierarchical Language Models for XML Component Retrieval », *in* N. Fuhr, M. Lalmas, S. Malik (eds), *Advances in XML Information Retrieval*, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 224-237, 2005.

Ounis I., Amati G., Plachouras V., He B., Macdonald C., Lioma C., « Terrier: A High Performance and Scalable Information Retrieval Platform », *Proceedings of ACM SIGIR'06 Workshop on Open Source Information Retrieval (OSIR 2006)*, 2006.

Piroi F., Lupu M., Hanbury A., « Overview of CLEF-IP 2013 Lab », *in* P. Forner, H. Müller, R. Paredes, P. Rosso, B. Stein (eds), *Information Access Evaluation. Multilinguality, Multimodality, and Visualization*, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 232-249, 2013.

Piroi F., Lupu M., Hanbury A., Magdy W., Sexton A., Filippov I., « CLEF-IP 2012: Retrieval experiments in the intellectual property domain », *CEUR Workshop Proceedings*, 01, 2012.

Risch J., Alder N., Hewel C., Krestel R., « PatentMatch: A Dataset for Matching Patent Claims Prior Art », 2020.

Robertson S., Zaragoza H., Taylor M., « Simple BM25 Extension to Multiple Weighted Fields », *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, CIKM '04, Association for Computing Machinery, New York, NY, USA, p. 42–49, 2004.

Sheetrit E., Kurland O., « Cluster-Based Focused Retrieval », *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, Association for Computing Machinery, New York, NY, USA, p. 2305–2308, 2019.

Sheetrit E., Shtok A., Kurland O., « A Passage-Based Approach to Learning to Rank Documents », 2019.

Verbyst D., Génération de documents virtuels par intégration de relations entre documents structurés pour la recherche d'information, Theses, Université Joseph-Fourier - Grenoble I, October, 2008.

Wilhelm-Stein T., Kürsten J., Eibl M., « Chemnitz at CLEF IP 2012: Advancing Xtrieval or a Baseline Hard to Crack. », 01, 2012.

Xue X., Croft W. B., « Automatic Query Generation for Patent Search », *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, Association for Computing Machinery, New York, NY, USA, p. 2037–2040, 2009.

Zhai C., Lafferty J., « A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval », *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, Association for Computing Machinery, New York, NY, USA, p. 334–342, 2001.