Kinship Contextualization: Utilizing the Preceding and Following Structural Elements

Muhammad Ali Norozi
Dept. of Computer and Information Science
Norwegian University of Science and Technology
Trondheim, Norway
mnorozi@idi.ntnu.no

Paavo Arvola
School of Information Sciences
University of Tampere
Tampere, Finland
paavo.arvola@uta.fi

ABSTRACT

The textual context of an element, *structurally*, contains traces of evidences. Utilizing this context in scoring is called contextualization. In this study we hypothesize that the context of an XML-element originated from its *preceding* and *following* elements in the sequential ordering of a document improves the quality of retrieval. In the tree form of the document's structure, *kinship* contextualization means, contextualization based on the horizontal and vertical elements in the *kinship tree,* or elements in closer to a wider structural kinship. We have tested several variants of kinship contextualization and verified notable improvements in comparison with the baseline system and gold standards in the retrieval of focused elements.

Categories and Subject Descriptors

H.3.3 [**Info. Search and Retrieval**]: Search process

Keywords

XML retrieval, Schema agnostic search, Contextualization

1. INTRODUCTION

Contextualization [3] is a mechanism which makes possible the retrieval of items with varying length in textual content, as the size of elements varies with the level in the hierarchy (see Figure 1); the leaf element or elements on low levels of hierarchy have potentially less textual evidences than their ancestors. The scant textual evidence in the small text units, such as paragraphs, are augmented with information obtainable from the context surrounding them.

The potential of contextualization has been revealed before in several intuitive settings [1–3, 5–8]. In the existing studies, context of elements in focused retrieval has been mainly referred to the ancestor elements. In addition to the hierarchical order or the ancestor elements, documents also have an established sequential ordering (paragraph 1 comes prior to paragraph 2 and hence are siblings in the structural tree (which we refer to as the *kinship tree*, see Figure 1), in the documents hierarchical structure). In this study the elements in the kinship tree, in the document's sequential order are considered to be the context - the kinship context. The proposed models are experimentally validated using the semantically annotated Wikipedia XML collection using INEX [9] evaluation measures. The results obtained, on focused retrieval task (INEX), exhibit clear improvements over the best submitted runs at INEX 2009, and over a strong and competitive baseline system – itself based on data fusion over all INEX 2009 submitted runs (Section 3).

Summarizing, the contributions of this study include:

- Contextualization utilizing the nodes in *kinship* relationship (Figure 1), in the hierarchical structure of documents, with random walks as a theoretically sound foundation (Section 2.1).
- Developing a competitive focused retrieval system baseline based on data fusion and constructing a test setting for evaluating the retrieval of small textual units, i.e., focused retrieval (Section 3).
- Experimental validation and evaluation (Section 4) of the role of kinship contextualization on the large semantically annotated Wikipedia XML corpora [9] (Section 4).

2. CONTEXTUALIZATION

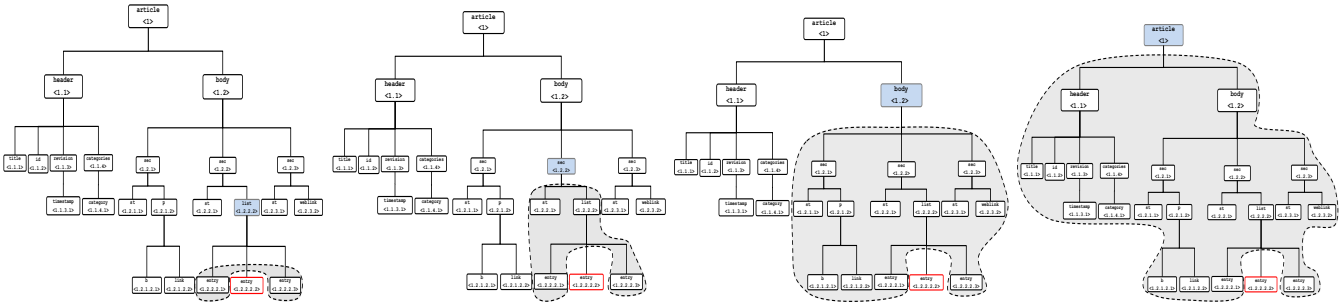
Contextualization is a re-scoring scheme, where the basic score, usually obtained from a full-text retrieval model, of a contextualized document or element is re-enforced by the weighted scores of the contextualizing documents or elements. We use random walks to induce a similarity structure over the documents based on the containment and reverse-containment relationships (element, sub-element and vice versa). Hence, these relationships (kinships) affect the weight each contextualizing element has in contextualization.

The premise is that *good context* (identified by random walk and contextualization model [6]) provides evidence that an element in focused retrieval is a good candidate for a posed query and therefore, the elements should be contextualized by their hierarchically similar elements in "kinship". Good context is an *evidence* that should be used to deduce that an element is a good candidate for the posed query.

2.1 Kinship Contextualization

In this section, we will show a formalism that can be used to materialize and then utilize the contextual evidences originated from the elements in the kinship tree, in the documents sequential order, for improving the retrieval effectiveness. Use of hierarchical information as a context has been studied before in different settings in XML retrieval [1, 3, 5, 8, 11]. In hierarchical contextualization the

SIGIR'13, July 28–August 1, 2013, Dublin, Ireland.



(a) $CR^{p}_{kinship}$ (b) $CR^{gp}_{kinship}$ (c) $CR^{ggp}_{kinship}$ (d) $CR^{a}_{kinship}$

Figure 1: Kinship tree taken from example in [6], a representative XML from the Wikipedia 2009 collection.

intrinsic structure within the XML document is employed. Kinship contextualization is both horizontal (siblings) and vertical (ancestors & descendants elements) but intrinsically non-hierarchical perspective of the hierarchical information. The representation of documents in XML aims to follow the established structure of documents, i.e., an academic book is typically composed of ⟨`chapters`⟩, ⟨`sections`⟩, ⟨`subsecs`⟩, ⟨`paragraph`⟩ and so on, structures. ⟨`chapter1`⟩ is followed by ⟨`chapter2`⟩ and within ⟨`chapter1`⟩, ⟨`section1`⟩ is followed by ⟨`section2`⟩. Elements ⟨`section1`⟩ and ⟨`section2`⟩ are siblings, and hence most likely, semantically related. The following element takes the concepts further from the preceding elements, and the preceding elements provide the basics or foundation for the following elements. Therefore, together in the document order, the *preceding* and *following* elements form a strong and connected perspective (the kinship context), surrounding the relevant information. This organization of document gives an intuitive starting point for manipulating text passages at the established hierarchy levels of text documents.

With contextualization from the preceding and following elements, we aim to rank higher an element in a good context (strong evidence in the kinship) than an identical element in a not so good context (less or no evidence in the kinship tree) within the document. In Figure 1 the ⟨`entry`⟩ element has the preceding ⟨`entry`⟩ and following ⟨`entry`⟩ as the kinship context. Hence one element can be viewed in context of its kinships, the elements preceding and following it.

In kinship contextualization, like the other types of contextualization [3, 6], the weight of the element is modified by the basic weights of its contextualizing elements. Each element in the context of the contextualized element, should possess an *impact* factor. An higher impact factor shows the importance of the contextualizing element and vice versa. The role and relation of contextualizing element are operationalized by giving the element a contextualizing weight. A contextualization vector is defined to capture the impact factor of each contextualizing element, and this contextualization vector is represented by a g function in Equation 1.

The important research question here is: how far wide (siblings) and how deep (direct parent or siblings at ancestral level), the element's kinship context help to improve retrieval effectiveness? Arvola et al. (2011) [3], in their horizontal contextualization approach, used a weight array, which follows a zero centred parabola function – the impact of contextualization is adjusted with a help of different set of parameters. The weight of contextualizing element is assumed to be the function of distance, hence the weight ought to be lower the further away the contextualizing element is from the contextualized element. Instead of employing the weight function, to find the impact of each of the units in the context, we propose here, a generalized mechanism based on the *Markovian* random walk principle [6] and the kinship contextualization.

2.1.1 Random Walk

The tree-structure of the XML document (Figure 1) is considered as a graph. Myriad of random surfers traverse the XML graphs. In particular, at any time step a random surfer is found at an element and either (a) makes a next move to the sub-element of the existing element by traversing the containment edge, or (b) makes a move to the parent-element of the existing element, or (c) jumps randomly to another element in the XML graph. As the time goes on, the expected percentage of surfer at each node converges to a limit the dominant eigenvector of the XML graph. This limit provides the impact or strength of each element in the context of the contextualized element in the form of g function. In kinship contextualization, we consider all the elements in the *kinship* of the contextualized element; where the contextualization vector g in this case, identifies the importance of each of the unit of context in kinship(Equation 1).

Contextualization model formulated in this way, is independent of the basic weighting scheme of the elements and it could be applied on the top of any query language and retrieval systems. We have applied the contextualization model on the top of the baseline system which is the result of fusion from the INEX 2009 officially submitted runs by the participants (see Section 3).

In the experiments we evaluated the retrieval effectiveness at different granularity levels. We mainly tested, retrieval effectiveness based on the element selection in focused retrieval task (using the INEX evaluation kit); a brief intuition is explained in Section 3.

2.1.2 Kinship Contextualization at different level

We have experimented with kinship contextualization at different levels of hierarchy. Kinship contextualization with elements in kinship from:

- direct parent, $CR^{p}_{kinship}$.
- grand parent, $CR^{gp}_{kinship}$.
- grand grand parent, $CR^{ggp}_{kinship}$.
- root, the ⟨`article`⟩ element, $CR^{a}_{kinship}$.

The four approaches listed above are pictorially shown in Figure 1.

2.1.3 Combination Function

The re-ranking function based on the random walk principle, described earlier, can be formally defined as follows:

$$CR(x, f, C_x, g^k) = BS(x) + f \cdot \frac{\sum_{y \in C_x} BS(y) \cdot g^k(y)}{\sum_{y \in C_x} g^k(y)} \quad (1)$$

where

- $BS(x)$ is the basic score of contextualized element x (text-based score, e.g., $tf \cdot ief$)
- f is a parameter which determines the weight of the context in the overall scoring.
- C_x is the kinship context surrounding the contextualizing element x, i.e., $C_x \subseteq kinships(x)$, $\subseteq$, because only the context containing the query terms are considered.
- $g^k(y)$ is the generalized contextualization vector based on random walk, which gives the authority weight (the impact) of y, the contextualizing elements (kinships) of x in XML graph. Similar interpretation is used in our earlier studies [6]

3. TEST SETTINGS & FUSION BASELINE

We test our approach using the Wikipedia collection containing 2.66 million semantically annotated XML documents (50, 7 Gb) and 68 related topics provided by the INEX 2009 ad-hoc track [4]. The reason for using the INEX 2009 test topics (instead of 2010) is the larger variety of elements in the participants' results which was due to the existence of the thorough task. In order to get the best possible baseline, we performed a data fusion based on sum of normalized scores (CombSUM) [10]. The element scores (for each run per topic) were normalized for the fusion as follows:

$$score_x = \frac{score_x - min(scores)}{max(scores) - min(scores)} \quad (2)$$

where $max(scores)$ and $min(scores)$ denote the maximal and minimal scores respectively.

We used all the 98 INEX 2009 runs delivering correct element result lists as component systems for the fusion. The 2009 runs have the largest variety in the results for the fusion in comparison to other years of the initiative. Unfortunately, most of the participants (56) did not report any real element scores, because the INEX evaluation did not require that information. For those systems an artificial score was given for each element based on their reciprocal rank, before the normalization. In other words, the first element in the result list was given a score 1, the second 1/2, third 1/3, fourth 1/4 and so on.

The focused task in INEX ad-hoc track is to retrieve most focused elements satisfying an information need without overlapping elements. An overlapping result list means that the elements in the result list may have a descendant relationship with each other and share the same text content. For instance, in Figure 1 the entry element ⟨`1.2.2.2.1`⟩ and the ⟨`sec`⟩ element ⟨`1.2.2`⟩ are overlapping. In this study we are following the focused approach, considering a result list where only one of the overlapping elements from each branch is selected. This means that including the ⟨`sec`⟩ element in the results would mean excluding the entry element in the results or vice versa.

The fused result list contains all the elements delivered by the 98 component systems. This comprehensive result list contains overlapping elements. In order to remove the overlap, we basically selected elements having the highest score from each branch. However, many participants returned runs having full-articles only, which led to full-article bias in the fusion results. Therefore, we made a deliberate choice to exclude full-articles in the results, following a more focused retrieval strategy. The result lists were measured using the official INEX evaluation metrics and software for the focused task [4].

Contextualization and the fusion approach as scoring methods, however, do not take any stand on which elements should be selected from each branch. Thus we perform a structural fusion, where we take the element level selection from the baseline run and subsequently re-rank the elements of the baseline run. For instance (in Figure 1) if the baseline run suggests the ⟨`body`⟩ element, we select that one, not the ⟨`list`⟩ element beneath, regardless of their mutual ranking in the full list.

4. EXPERIMENTAL EVALUATION

The hierarchical structure of XML documents are captured using the dewey encoding scheme (as shown in the Figure 1). This way each element in the document possess a unique index within the document, and together with document's unique id, this becomes unique for the entire collection. The tree structure of XML documents are converted into a matrix, and random walk is performed on this matrix, as it is described in detail, in our earlier work [6]. The contextualization vector g^k from Equation 1 is computed off-line for each and every XML document in the Wikipedia collection. This suggests that computing g^k vector is feasible for a reasonably large XML document collections. At the query time, the scores from g^k vector and the basic scores are combined to produce an overall ranking score, using Equation 1.

We have experimented with all the four variants of kinship contextualization (Section 2.1) and compared them against the different baseline systems, (Table 1, sorted on *i*nterpolated *p*recision at recall 0.01, $iP[0.01]$). The runs in Table 1 are among the best runs submitted at INEX 2009 ad-hoc track, focused retrieval task.

Run ID	MAiP	iP[0.00]	iP[0.01]	iP[0.05]	iP[0.10]
CombSUM Fusion	**.3396**	**.7577**	**.7273**	**.6539**	**.6021**
UWFerBM25F2	.1854	.6797	.6333	.5006	.4095
I09LIP6Okapi	.3000	.6244	.6141	.5823	.5290
UJM_15525	.2890	.6241	.6060	.5742	.4921
UamsFSecs2dbi100CA	.1928	.6328	.5997	.5141	.4647
BM25BOTrangeFOC	.2912	.6049	.5992	.5619	.5057
Spirix09R001	.2865	.6081	.5903	.5342	.4979
LIG-2009-focused-1F	.2702	.5861	.5853	.5431	.5055

Table 1: Retrieval statistics for baseline systems. CombSUM fusion run is the best (statistically significant on all measures at $p < 0.01$, 1-tailed t-test).

In the combination function given, the contextualization force has to be parametrized. For the proposed contextualization model, we tuned the contextualization force and report the values leading to best overall performance. In our parametrization process we found the optimal values of contextualization force f (from Equation 1) lies in the range: ($f \in \{3.25, 3.50, 3.75, 4.00, 4.25, 4.50\}$). These optimal values for f are obtained by using cross-validation technique. We did 68-fold cross-validation (or complete cross-validation in our case) - by randomly partitioning the collection into 68

training and test samples based on the number of assessed topics. Of the 68 samples, a single sample is retained as the validation set for testing, and remaining 67 samples are used as training set. The cross-validation process is repeated 68 times (for each fold), with each of 68 samples used exactly only once as validation set. These 68 independent or unseen samples are then combined to produce a single or a set of estimations for parameter f.

Method	f	MAiP	iP[0.00]	iP[0.01]	iP[0.05]	iP[0.10]
CombSUM Fusion	–	**.3396**	.7577	.7273	.6539	.6021
UWFerBM25F2	–	.1854	.6797	.6333	.5006	.4095
$CR^{p}_{kinship}$	3.25-4.5	.2949*	.7357*	.6971^{+}	.6580*	.6066*
$CR^{gp}_{kinship}$	3.25-4.5	.3034*	.7746$^{\triangle}$*	.7308$^{\triangle}$*	.6945$^{\triangle}$*	.6457$^{\triangle}$*
$CR^{ggp}_{kinship}$	3.25-4.5	.3158*	**.8125**$^{\blacktriangle}$*	**.7552**$^{\blacktriangle}$*	**.7145**$^{\blacktriangle}$*	**.6572**$^{\blacktriangle}$*
$CR^{a}_{kinship}$	3.25-4.5	.3049*	**.8046**$^{\blacktriangle}$*	.7490$^{\triangle}$*	.6993$^{\triangle}$*	.6499$^{\triangle}$*

Table 2: Ret. performance for focused retrieval $^{\blacktriangle}$* = stat. significant than both the CombSUM Fusion and UWFerBM25F2 at p < 0.01, and $^{\triangle+}$ = stat. significant at p < 0.05 respectively.

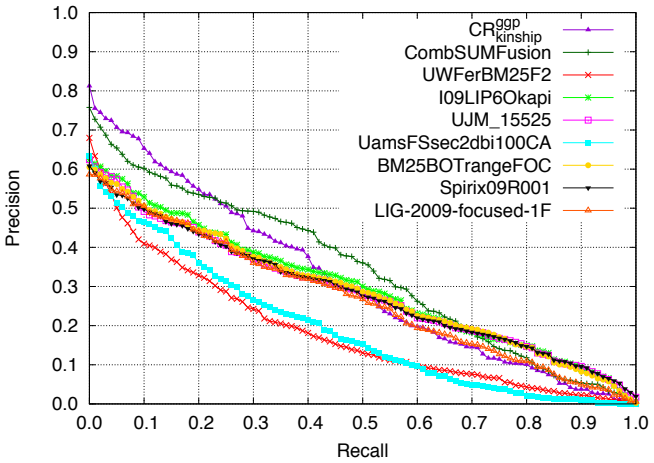


Figure 2: Precision - recall performance against baselines and best INEX 2009 submitted runs.

Table 2 and Figure 2 show the overview of the retrieval performance of our approaches against the baselines for the focused retrieval task. The proposed contextualization model improves the performance over the baselines. The improvements are found to be statistically significant (1-tailed t-test at $p < 0.01$ and $p < 0.05$) on iP and $MAiP$ measures.

The best overall results among the proposed methods are obtained with $CR^{ggp}_{kinship}$ and $CR^{a}_{kinship}$, in terms of best iP[0.01] values (early precision). The kinship context from the hierarchical structure of documents, employed in contextualization, indeed improves the retrieval effectiveness, and the improvements are in-line with theoretical anticipations.

5. CONCLUSIONS AND DISCUSSION

Contextualization is a re-ranking model, utilizing the context of the relevant retrievable unit, for improving the overall retrieval. We have presented an exploratory study into the use of context from elements in kinship in the hierarchical structure of information, to improve retrieval performance on focused retrieval tasks. We looked at context from document's sequential ordering, which we call the kinship context. Hence, we hypothesized that context gathered from the kinships, "horizontally" and "vertically" from the graph structure of document, influences the retrieval effectiveness. Experiments have validated the hypothesis that utilizing the kinship context this way actually enhances the retrieval of information in focused retrieval task. The results obtained are in-line with the earlier work on contextualization [1, 3, 5–8]. However, none of the existing works consider the kinship context, as a source of contextual evidence.

The approaches presented are generic and can be applied to different test collections and baseline systems. Evidence are collected in a systematic way, from the surroundings, the kinship context of the element to be ranked. XML documents are used as a sample case of semi-structured documents, these documents have hierarchical structure, which is often represented in a form of tree. However, the approaches could also be applicable for other generic structured (or semi-structured) test collections (e.g., Linked Data, RDF, etc.), where the structure may be represented as a general graph (with cycles). The proposed methods are particularly suited for collections that carry more types of evidence than just textual information.

6. REFERENCES

[1] P. Arvola, M. Junkkari, and J. Kekäläinen. Generalized Contextualization Method for XML Information Retrieval. In *Proc. of the 14th ACM CIKM*, pages 20–27. ACM, 2005.

[2] P. Arvola, J. Kekäläinen, and M. Junkkari. The Effect of Contextualization at Different Granularity Levels in Content-oriented XML Retrieval. In *Proc. of the 17th ACM CIKM*, pages 1491–1492. ACM, 2008.

[3] P. Arvola, J. Kekäläinen, and M. Junkkari. Contextualization Models for XML Retrieval. *Info. Processing & Management*, pages 1–15, 2011.

[4] S. Geva, J. Kamps, M. Lethonen, R. Schenkel, J. Thom, and A. Trotman. Overview of the inex 2009 ad hoc track. *Focused Retrieval and Evaluation*, pages 4–25, 2010.

[5] Y. Mass and M. Mandelbrod. Component Ranking and Automatic Query Refinement for XML Retrieval. *Advances in XML IR*, pages 1–18, 2005.

[6] M. A. Norozi, P. Arvola, and A. P. de Vries. Contextualization using hyperlinks and internal hierarchical structure of wikipedia documents. In *Proc. of the 21st ACM CIKM*, pages 734–743. ACM, 2012.

[7] P. Ogilvie and J. Callan. Hierarchical Language Models for XML Component Retrieval. *Advances in XML IR*, pages 269–285, 2005.

[8] G. Ramirez Camps. *Structural Features in XML Retrieval*. PhD thesis, SIKS, the Dutch Research School for Information and Knowledge Systems., 2007.

[9] Schenkel, R. and Suchanek, F.M. and Kasneci, G. YAWN: A Semantically Annotated Wikipedia XML Corpus. *Proc. of GIFachtagung für Datenbanksysteme in Business Technologie und Web BTW2007*, 103 (Btw):277–291, 2007.

[10] J. A. Shaw and E. A. Fox. Combination of multiple searches. In *The 2nd TREC*. Citeseer, 1994.

[11] B. Sigurbjörnsson, J. Kamps, and M. De Rijke. An Element-based Approach to XML Retrieval. In *INEX 2003 Workshop Proc.*, pages 19–26, 2004.