Resolving XML Semantic Ambiguity

Technical Report – XSDF-TR-2014

Nathalie Charbel
LE2I Lab. UMR-CNRS, University of Bourgogne (UB), 21078 Dijon, France
nathaliecharbel@gmail.com

Joe Tekli
ECE Dept., Lebanese American University (LAU), 36 Byblos, Lebanon
joe.tekli@lau.edu.lb

Richard Chbeir
LIUPPA Lab., University of Pau & Adour Countries (UPPA), 64600 Anglet, France
richard.chbeir@univ-pau.fr

Gilbert Tekli
NOBATEK,
67 Rue de Mirambeau
64600 Anglet, France
gtekli@gmail.com

ABSTRACT

XML semantic-aware processing has become one of the central issues in Web data management, data processing, and information retrieval. While XML data is semi-structured, yet it remains prone to lexical ambiguity, and thus requires dedicated semantic analysis and sense disambiguation processes to assign well-defined meaning to XML elements and attributes. This becomes crucial in an array of applications ranging over semantic-aware query rewriting, semantic document clustering and classification, schema matching, as well as blog analysis and event detection in social networks and tweets. Most existing approaches in this context: i) ignore the problem of identifying ambiguous XML nodes, ii) only partially consider their structural relations/context, iii) use syntactic information in processing XML data regardless of the semantics involved, and iv) are static in adopting fixed disambiguation constraints thus limiting user involvement. In this paper, we provide a new XML Semantic Disambiguation Framework titled *XSDF* designed to address each of the above motivations, taking as input: an XML document and a general purpose semantic network, and then producing as output a semantically augmented XML tree made of unambiguous semantic concepts. Experiments demonstrate the effectiveness of our approach in comparison with alternative methods.

This technical report contains proofs, computation examples, and experimental results adding to the contributions of the main paper.

CONTENTS

I. Proofs of Propositions and Lemmas

Proposition 1: The ambiguity degree of an XML node x in tree T increases when the number of senses of $x.\ell$ is high in the reference semantic network SN, or else it decreases such that:

$$\text{Amb}_{Polysemy}(x.\ell, SN) = \frac{senses(x.\ell) - 1}{\text{Max}(senses(SN)) - 1} \in [0,1] \quad \textbf{(1)}$$

where $Max(senses(SN))$ is the maximum number of senses of a word/expression in SN (e.g., in WordNet 2.1 [14], $Max_{polysemy}$ = 33, for the word *"head"*) □

Proof of Prop. 1: Based on formula 1, $Amb_{Polysemy}$ varies as follows:

- The minimum value $Amb_{Polysemy} = 0$ is obtained when $x.\ell = 1$, i.e., ℓ has only one sense (e.g., *"first name"* in WordNet), i.e., x is without ambiguity: it always refers to the same meaning.
- The maximum value $Amb_{Polysemy} = 1$ is obtained when $x.\ell = Max(polysemy(SN))$.
- When $polysemy(x.\ell)$ increases/decreases, $Amb_{Polysemy}$ follows accordingly such that $Amb_{Polysemy} \in [0, 1]$ □

Proposition 2: The ambiguity degree of an XML node x in tree T increases when the distance in number of edges between x and $R(T)$ is low, or else it decreases such that:

$$\text{Amb}_{Depth}(x, T) = 1 - \frac{x.d}{\text{Max}(depth(T))} \in [0,1] \quad \textbf{(2)}$$

where $Max(depth(T))$ is the maximum depth in T □

Proof of Prop. 2: Following formula 2, Amb_{Depth} varies as follows:

- The maximum value $Amb_{Depth} = 1$ is obtained when $x.d = 0$, i.e., when $x = R(T)$.
- The minimum value $Amb_{Depth} = 0$ is obtained when $x.d = Max(depth(T))$, i.e., when x is one of the farthest nodes from $R(T)$: one of the deepest (most specific) leaf nodes in T's hierarchy.
- When $x.d$ increases/decreases, Amb_{Depth} follows inversely such that $Amb_{Depth} \in [0, 1]$ □

Proposition 3: The ambiguity degree of an XML node x in tree T increases when the number of children nodes of x having distinct node labels, designated as $\overline{x.f}$, is low, or else it decreases:

$$\text{Amb}_{Density}(x, T) = 1 - \frac{\overline{x.f}}{\text{Max}(\overline{fan\text{-}out(T)})} \in [0,1] \quad \textbf{(3)}$$

where $Max(\overline{fan\text{-}out(T)})$ is the maximum number of children nodes with distinct node labels in T. We identify this factor as node density factor to distinguish it from traditional node fan-out: number of children nodes (regardless of label, cf. **Error! Reference source not found.**) □

Proof of Prop. 3: Following formula 3, $Amb_{Density}$ varies as follows:

- The maximum value $Amb_{Density} = 1$ is obtained when $\overline{x.f} = 0$, i.e., when x is a leaf node and does not have any children nodes (to provide hints concerning x's meaning).

– The maximum value $Amb_{Density}$= 0 is obtained when $\overline{x.f}$ = *Max*(*fan-out*(*T*)), i.e., when *x* has the largest number of children nodes with distinct labels in *T*. In other words, it has the highest possible number of hints regarding its meaning in *T*.
– When $\overline{x.f}$ increases/decreases, $Amb_{Density}$ follows inversely such that $Amb_{Density} \in [0, 1]$ □

Lemma 1: The ambiguity degree measure *Amb_Deg* in **Error! Reference source not found.** varies in accordance with Propositions 1-3, and conforms to Assumptions 1-4 (refer to main paper) □

Proof of Lemma 1: Following formula 4 in the main paper, *Amb_Deg* varies as follows:

– The minimum value *Amb_Deg* = 0 is obtained when the label of node *x* has only one possible sense, i.e., when $Amb_{Polysem}(x.\ell, SN)$ = 0, regardless of its depth and density factors, and regardless of parameter weights (Assumption 4).
– The maximum value *Amb_Deg* = 1 is obtained when: i) $x.\ell$ has the maximum number of senses in *SN* (e.g., 33 senses in WordNet [16]), ii) *x* is the root node of *T*, *x.d* = 0, and iii) *x* has the minimum number of children nodes with distinct labels in *T*, $\overline{x.f}$ = 0, regardless of w_{Depth} and $w_{Density}$ parameter values (Assumptions 1-3, and Lemmas 1-3)
– The value of *Amb_Deg* increases with: i) the increase in $x.\ell$'s polysemy in *SN*, ii) the decrease in *x*'s depth, and iii) the decrease in *x*'s density in the XDT. Inversely, the value of *Amb_Deg* decreases with: i) the decrease in $x.\ell$'s polysemy, ii) the increase in *x*'s depth, and iii) the increase in *x*'s density (Assumptions 1-3, and Lemmas 1-3).

Lemma 2: The context vector weight measure $w_{\overrightarrow{V_d(x)}}(\ell_r)$ in **Error! Reference source not found.** varies in accordance with Assumptions 5 and 6 (refer to main paper for definition and assumptions) □

Proof of Lemma 2: Based on formula 5 in the main paper, $w_{\overrightarrow{V_d(x)}}(\ell_r)$ varies as follows:

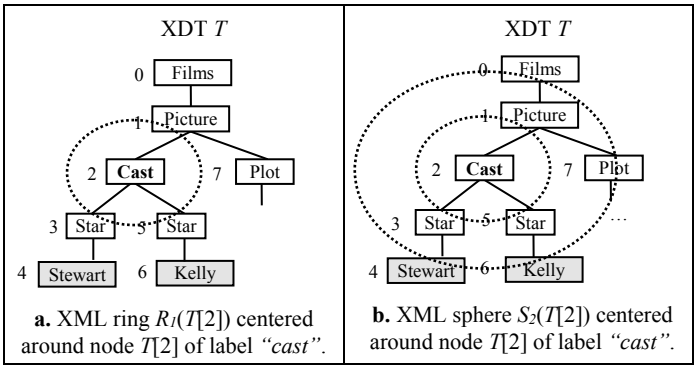
– $w_{\overrightarrow{V_d(x)}}(\ell_r)$ increases with $Freq(\ell_r, S_d(x))$, which increases with $Struct(x_i, S_d(x))$:
  - The value of $Struct(x_i, S_d(x))$ is inversely proportional to the distance between the target node *x* and context node x_i (following Assumption 1).
    - The minimum value $Struct(x_i, S_d(x)) = \frac{1}{d+1}$ is reached when $x_i \in R_d(x)$ where $R_d(x)$ is the outer-most ring in sphere $S_d(x)$.
    - The maximum value $Struct(x_i, S_d(x))$=1 is reached when processing the target node itself, i.e., $x_i = x$.
  - The value of $Freq(\ell_r, S_d(x))$ is proportional to the number of occurrences of nodes having the same label $x_i.\ell = \ell_r$ (following Assumption 2).
    - The minimum value $Freq(\ell_r, S_d(x)) = \frac{1}{d+1}$ is reached when there is only one context node having label ℓ_r and occurring on the outer-most ring $R_d(x)$ of the sphere neighborhood, more formally:
      $\exists\, x_i \in S_d(x) \;/\; (x_i.\ell = \ell_r) \wedge (x_i \in R_d(x)) \wedge (\forall\, x_j \in S_d(x) \;/\; x_j.\ell \neq \ell_r)$
    - The maximum value of $Freq(\ell_r, S_d(x)) = \frac{|S_d(x)|+1}{2}$ is reached when all context nodes have the same label ℓ_r and appear on the inner-most ring $R_1(x)$ of the sphere neighborhood, more formally:
      $\forall\, x_i \in S_d(x) \;/\; (x_i.\ell = \ell_r) \wedge (x_i \in R_1(x))$. Here, $Freq(\ell_r, S_d(x)) = 1 + \frac{1}{2} \times (|S_d(x)| - 1) = \frac{|S_d(x)|+1}{2}$

Consequently:

– The minimum value $w_{\overrightarrow{V_d(x)}}(\ell_r) = 0$ is obtained when no nodes of label ℓ_r occur in the sphere context of target node *x*.
– The maximum value $w_{\overrightarrow{V_d(x)}}(\ell_r) = 1$ is obtained when maximum frequency is obtained, since the weight score is normalized using maximum frequency, i.e., $\frac{|S_d(x)|+1}{2}$ □

II. Computation Examples

Consider XML document trees in Figure 1 (reported from the main paper), where the XML sphere $S_2(T[2])$ is centered around node *T*[2] of label *"cast"* with radius 2 consists of: ring $R_1(T[2])$ of radius 1 comprises nodes *T*[1] (*"picture"*), *T*[3] (*"star"*) and *T*[5] (*"star"*), and ring $R_2(T[2])$ of radius 2 comprises nodes *T*[0] (*"Films"*), T[4] (*"Stewart"*), T[6] (*"Kelly"*), and *T*[7] (*"Plot"*).



a. XML ring $R_1(T[2])$ centered around node *T*[2] of label *"cast"*.

b. XML sphere $S_2(T[2])$ centered around node *T*[2] of label *"cast"*.

Figure 1. Sample XML (ring and) sphere neighborhoods**.**

	Cast	*Picture*	*Star*
$\overrightarrow{V_1(T[2])}$	0.4	0.2	0.4

	Cast	*Picture*	*Star*	*Films*	*Stewart*	*Kelly*	*Plot*
$\overrightarrow{V_2(T[2])}$	0.25	0.1667	0.3334	0.0835	0.0835	0.0835	0.0835

Figure 2. Sample sphere context vectors based on the sphere neighborhoods in Figure 1.

Considering $\overrightarrow{V_1(T[2])}$ in Figure 2:

- $w_{\overrightarrow{V_1(\text{“Cast”})}}(\text{“Cast”}) = \frac{2\times(1)}{(4)+1} = 0.4$ given that: i) *“Cast”* is the label of $S_1(T[2])$'s target (center) node $T[2]$, i.e., $Struct\left(T[2], S_1(T[2])\right)$=1, ii) $T[2]$ is the only node occurrence of label *“cast”* in $S_1(T[2])$, i.e., $Freq\left(\text{“Cast”}, S_1(T[2])\right) = 1$, and iii) $|S_1(T[2])| = 4$.

- $w_{\overrightarrow{V_1(\text{“Cast”})}}(\text{“Picture”}) = \frac{2\times(0.5)}{(4)+1} = 0.2$ given: i) $S_1(T[2])$ contains one node $T[1]$ of label *“Picture”* positioned at distance = 1 from the target node $T[2]$, i.e., $Struct\left(T[3], S_1(T[2])\right)$ = 1-$\frac{1}{2}$= 0.5, ii) $T[2]$ is the only node occurrence of label *“Picture”* in $S_1(T[2])$, i.e., $Freq\left(\text{“Picture”}, S_1(T[2])\right) = 0.5$, and iii) $|S_1(T[2])| = 4$.

Likewise, considering vector $\overrightarrow{V_2(T[2])}$: $w_{\overrightarrow{V_2(\text{“Cast”})}}(\text{“Star”}) = \frac{2\times(0.6667+0.6667)}{(7)+1} = 0.3335$ given: i) $S_1(T[2])$ contains two nodes $T[3]$ and $T[5]$ of label *“Star”* at distance = 1 from the target node $T[2]$, i.e., $Struct\left(T[3], S_1(T[2])\right)$= 1-$\frac{1}{3}$= 0.6667, ii) $T[3]$ and $T[5]$ are the only node occurrences of label *“Star”* $S_1(T[2])$, i.e., $Freq\left(\text{“Star”}, S_1(T[2])\right) = 0.6667 + 0.6667 = 1.3334$ and iii) $|S_2(T[2])| = 7$.

III. Disambiguation Algorithms

Algorithm *XSD*$_{Concept}$

Input: x, d // Target XML node *x* and sphere neighborhood radius *d*
T // Source XML tree
$\overline{SN}$ // Reference weighted semantic network
Output: c // Semantic concept representing the sense (meaning) of $x.\ell$

```
Begin                                                                          1
  Generate Sd(x) ∈ T and V_d(x)   // Context vector of x in T                  2
  For each sp ∈ SN̄ → x.ℓ   // For each sense of the target node label         3
  {                                                                            4
    For each xi ∈ Sd(x)   // Processing senses of context nodes                5
    {                                                                          6
      For each s^i_j ∈ SN̄ → xi.ℓ  { Sim_Score(s^i_j) = Sim(sp, s^i_j) }        7
      Max_Score(xi.ℓ) = Max_{s^i_j → xi.ℓ} ( Sim_Score(s^i_j) )                8
    }                                                                          9
    Concept_Score (sp,) = Σ_{xi ∈ Sd(x)} Max_Score(xi.ℓ) × w_{Vd(x)}(xi.ℓ) / |Sd(x)|   10
  }                                                                            11
  c = sk / Concept_Score(sk) = Max_{sr → x.ℓ} ( Concept_Score(sp) )            12
  Return c                  // Sense (concept) with maximum score             13
End                                                                            14
```

Figure 3. Algorithm *XSD*$_{Concept}$ for concept-based XML Semantic Disambiguation.

Algorithm *XSD*$_{Context}$

Input: x, d // Target XML node *x* and sphere neighborhood radius *d*
T // Source XML tree
SN // Reference semantic network
Output: c // Semantic concept representing the sense (meaning) of $x.\ell$

```
Begin                                                                          1
  Generate Sd(x) ∈ T and V_d(x)   // Context vector of x in T                  2
  For each sp ∈ SN → x.ℓ   // For each sense of the target node label         3
  {                                                                            4
    Generate Sd(sp) ∈ SN and V_d(sp)   // Context vector of sp in SN           5
    Context_Score(sp) = (V_d(x) . V_d(sp)) / (||V_d(x)|| × ||V_d(sp)||)   // Context similarity   6
  }                                                                            7
  c = sk / Context_Score(sk) = Max_{sr → x.ℓ} ( Context_Score(sp) )            8
  Return c                  // Sense (concept) with maximum score             9
End                                                                            10
```

Figure 4. Algorithm *XSD*$_{Context}$ for context-based XML Semantic Disambiguation.

IV. Evaluating XML Node Ambiguity

Detailed manual and system ambiguity ratings concerning the three extreme correlations scores (*maximum*, closest to *null*, and *minimum* scores) highlighted in bold in Table 1 (report from the main paper), are shown in Figures 5-8 respectively.

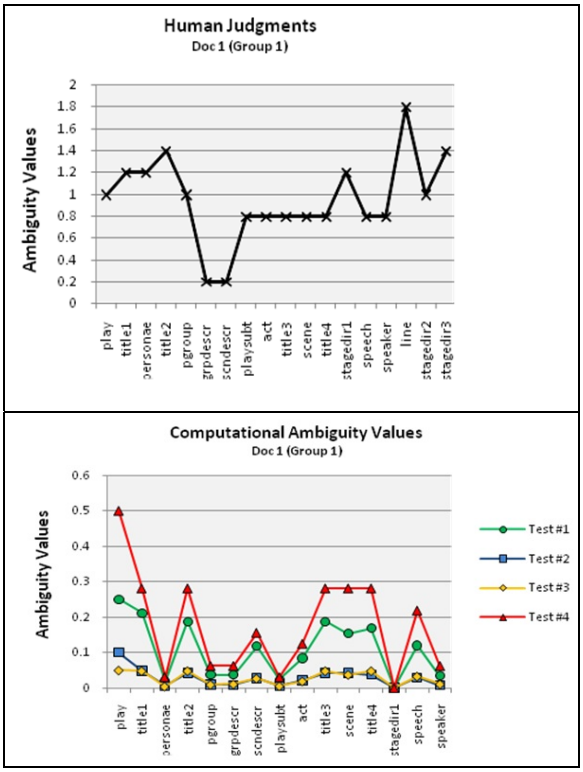


Figure 5. Manual and system generated average ambiguity degrees highlighting maximum correlation with documents of data-set 1 of Group 1. The *x* axis represents node labels (tag names and/or values) being disambiguated.

Table 1. Correlation between human ratings and system generated ambiguity degrees (reported from the main paper).

		Test #1 *All factors*	Test #2 *Polysemy*	Test #3 *Depth*	Test #4 *Density*
Group 1	Data-set 1	0.394	0.411	0.335	**0.439**
Group 2	Data-set 2	**0.017**	0.181	0.243	0.139
Group 3	Dataset 3	-0.087	-0.139	-0.071	-0.138
	Data-set 4	0.408	0.438	0.390	0.398
	Data-set 5	-0.184	-0.185	-0.131	-0.235
Group 4	Data-set 6	-0.284	-0.291	-0.243	-0.316
	Data-set 7	-0.177	-0.190	-0.254	-0.143
	Data-set 8	-0.119	-0.025	0.033	-0.156
	Data-set 9	-0.452	-0.301	-0.251	**-0.456**
	Data-set 10	-0.258	0.180	0.412	0.276

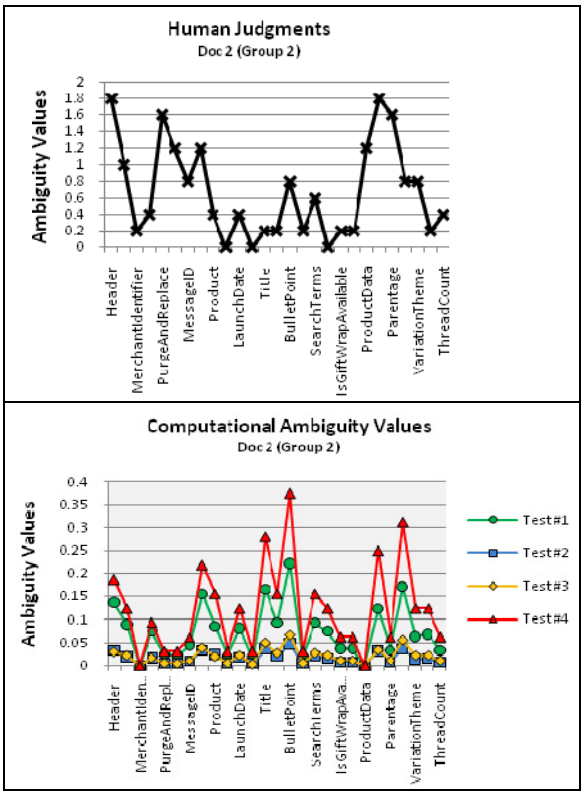


Figure 6. Manual and system generated average ambiguity degrees highlighting closest to zero (null) correlation with documents of data-set 2 of Group 2.

```
<?xml version="1.0"?>
<company>
    <cname id="id3_4">Informix</cname>
    <address>
        <street>123 6th Ave W</street>
        <city>Portland</city>
        <state>OR</state>
        <zip>54678</zip>
    </address>
    <personnel>
        <person>
            <name>
                <given>Fran</given>
                <family>Car</family>
            </name>
            <url>http://null</url>
        </person>
    </personnel>
</company>
```

```
<?xml encoding="ISO-8859-1"?>
<!ELEMENT company (address, cname,
                   personnel)>
<!ATTLIST comapny id ID #REQUIRED>
<!ELEMENT address (street, city, state, zip)>
<!ELEMENT personnel (person)+>
<!ELEMENT person (name, email?, url?)>
<!ELEMENT family (#PCDATA)>
<!ELEMENT given (#PCDATA)>
<!ELEMENT name (family?|given?)>
<!ELEMENT cname (#PCDATA)>
<!ELEMENT email (#PCDATA)>
<!ELEMENT street (#PCDATA)>
<!ELEMENT city (#PCDATA)>
<!ELEMENT state (#PCDATA)>
<!ELEMENT zip (#PCDATA)>
<!ELEMENT url (#PCDATA)>
```

Figure 7. Sample XML document from data-set 9 of Group 4, with corresponding grammar.

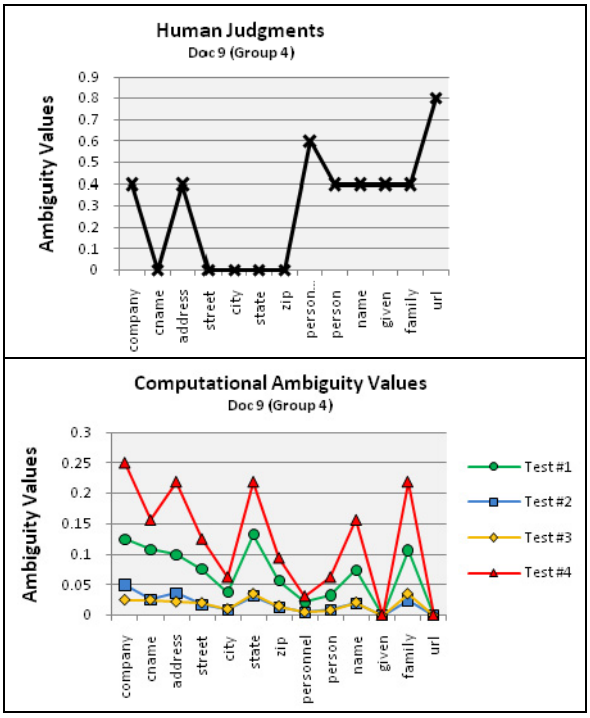


Figure 8. Manual and system generated average ambiguity degrees highlighting minimum correlation with documents of data-set 9 of Group 4 (Figure 7). The *x* axis represents node labels (tag names and/or data values).

V. Time Analysis

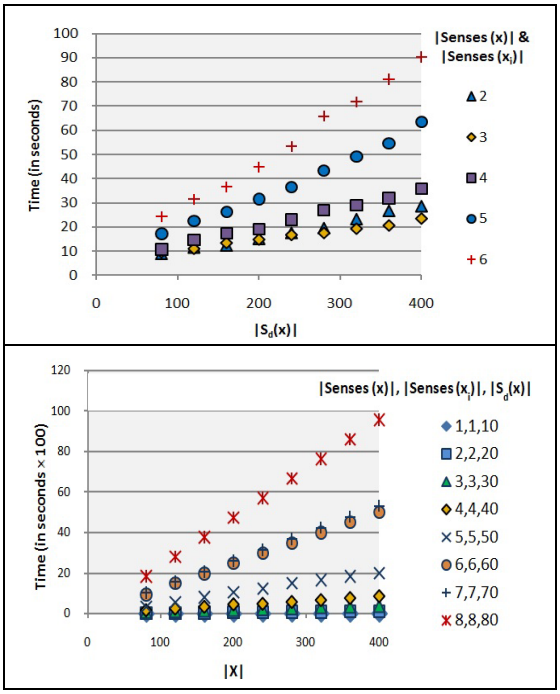


Figure 9. Timing results regarding our *concept-based* approach (when executing algorithm $XSD_{Concept}$, cf. Figure 3).

We evaluated the efficiency of our approach in terms of execution time. Results highlight the linear complexities of both our concept-based approach i.e., $O(|senses(x.\ell)| \times |S_d(x)| \times |senses(x_i.\ell)|)$ (Figure 9) and our context-based approach, i.e., $O(|senses(x.\ell)| \times (|S_d(x)| + |S_d(s_p)|)$ (Figure 10). Time is also linear w.r.t. the number of target nodes being disambiguate, designated as $|X|$. The complete experimental study (along with test documents and the detailed properties of XML nodes being disambiguated, e.g., label polysemy, node depth, density, etc.) is available online[1].

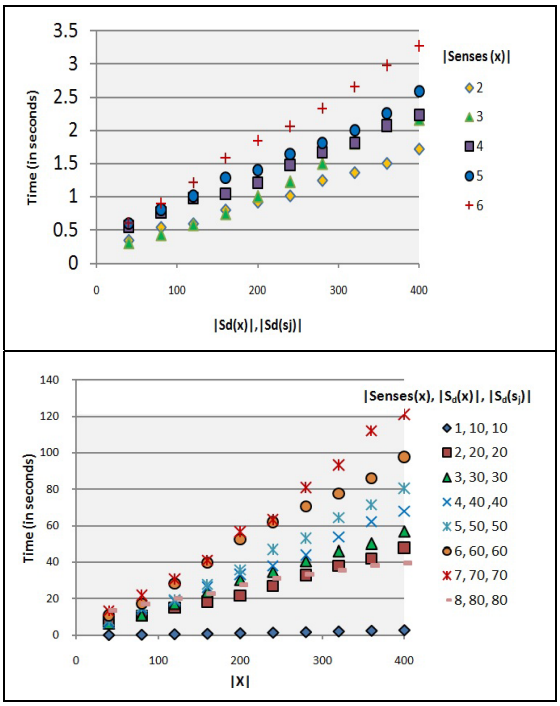


Figure 10. Timing results regarding our *context-based* approach (when executing algorithm $XSD_{Context}$, cf. Figure 4).

We have also compared the time complexity of our approach using different configurations, with one of its most recent predecessors. Results in Figure 11 show closely correlated and even reduced time results (depending on the configuration used, e.g., smaller context radiuses in XML document and/or in the reference semantic network). This means that our approach was able to produce improved disambiguation quality while preserving (and sometimes reducing) execution time levels in comparison with its alternatives[2].

[1] http://sigappfr.acm.org/Projects/XSDF/

[2] Note that we did not compare execution time with the *Versitile Structure Disambiguation* approach [2] since we were unable to acquire the system implementation from the authors. We used the authors' online version of the prototype, which is relatively slow due to network access, and thus could not use it to evaluate processing time.

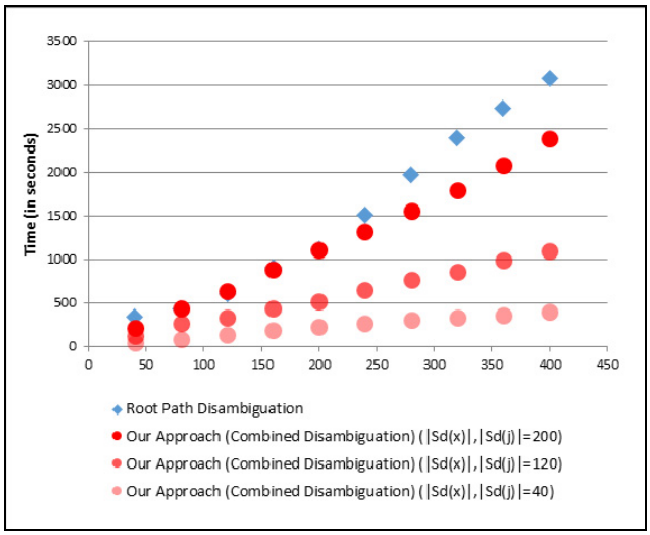


Figure 11. Comparing time results with existing *Root Path Disambiguation* [1] approach.

REFERENCES

[1] Tagarelli A. *et al.*, *Word Sense Disambiguation for XML Structure Feature Generation.* In Proc. of ESWC, 2009, pp. 143–157.

[2] Mandreoli F. *et al.*, *Versatile Structural Disambiguation for Semantic-Aware Applications.* In Proc. of Inter. CIKM Conf., 2005. pp. 209-216.