This is the author version of an article published as:

Woodley, Alan P. and Geva, Shlomo (2007) NLPX at INEX 2006. In Fuhr, Norbert and Lalmas, Mounia and Trotman, Andrew, Eds. Proceedings Comparative Evaluation of XML Information Retrieval Systems, 5th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2006 4518/2007, pages pp. 302-311, Dagstuhl Castle, Germany.

NLPX at INEX 2006

Alan Woodley, Shlomo Geva

School of Software Engineering and Data Communications, Faculty of Information Technology, Queensland University of Technology
GPO Box 2434, Brisbane, Queensland, Australia
{ap.woodley@student.qut.edu, s.geva@qut.edu.au}

Abstract. XML information retrieval (XML-IR) systems aim to better fulfil users' information needs than traditional IR systems by returning results lower than the document level. In order to use XML-IR systems users must encapsulate their structural and content information needs in a structured query. Historically, these structured queries have been formatted using formal languages such as NEXI. Unfortunately, formal query languages are very complex and too difficult to be used by experienced - let alone casual - users and are too closely bound to the underlying physical structure of the collection. INEX's NLP task investigates the potential of using natural language to specify structured queries. QUT has participated in the NLP task with our system NLPX since its inception. Here, we discuss the changes we've made to NLPX since last year, including our efforts to port NLPX to Wikipedia. Second, we present the results from the 2006 INEX track where NLPX was the best performing participant in the Thorough and Focused tasks.

1 Introduction

Information retrieval (IR) systems respond to users' queries with a ranked list of relevant results. In traditional IR systems these results are whole documents, but since XML documents separate content and structure XML-IR systems are able to return highly specific results that are lower than the document level. But, if users are going to take advantage of this capability in an operational (and possibly even in a laboratory setting) then they require an interface that is powerful enough to express their content and structural requirements, yet user-friendly enough that they can express their requirements intuitively.

Historically, XML-IR systems have used two types of interfaces: keyword based and formal query language based. Keyword based systems are user-friendly, but are unable to express the structural needs of the user. In comparison, formal query language-based interfaces are able to express users' structural needs (as well as their content needs) but are impractical for operational use since they too difficult to use — especially for casual users [7,14] — and are bound to the physical structure of the document. The purpose of INEX's natural language processing (NLP) track is to investigate a third interface option that encapsulates users' content and structural needs intuitively, in a natural language query (NLQ). Participants in the NLP track develop systems that translate natural languages queries to formal language queries (NEXI). The translated queries are executed on a single backend retrieval system (GPX) and their retrieval performance is compared amongst them themselves and with a baseline system consisting of manually constructed NEXI queries. The NLP task uses the same topics, documents and relevance assessments as the Ad-hoc task to enable comparison between participants in the NLP tracks as well as with participants in the Ad-hoc track (although, NLP participants process the topics' *description* elements rather than their *title* elements).

QUT has participated in the NLP task since its inception with its natural language interface NLPX. Here, we discuss the changes made to NLPX since last year and discuss its performance at this year's INEX. The changes subsume two parts. First, we outline the special connotations and templates added to NLPX that allow it to process a greater range of structured NLQs. Second, we discuss the process of porting NLPX from the IEEE collection to the Wikipedia collection. We also present the results from this the 2006 NLP track where NLPX was the best performing participant in the Thorough and Focused tasks.

2 Motivation

We have already outlined the motivations for an XML-IR natural language interface in our previous work [12,13]; however, for completeness we include them here. The motivations stem from weakness with current XML-IR interfaces.

The reason that keywords are unsuitable for XML-IR is that they can only contain users' content information need and not their structural information need. It has long been assumed that a user's information need will be better fulfilled if they specify their structural need, that is, the location within the document that contains their desired content. However, recent research has shown that this assumption may not be correct when considered over a number of queries and retrieval algorithms [9]. However, it is unclear if this outcome is because the specification of structural requirements does not assist retrieval at all or for other reasons (such as users not being able to correctly specify structural requirements, lack of meaningful structure within INEX"s previous IEEE document collection or an inability for existing XML-IR systems to handle structural requirements).

While formal languages are able to fully capture users' structural requirements, they too have problems. First, formal query languages are too difficult for both expert and casual users to correctly express their structural and content information needs. The difficulty that experts have in using formal languages has been recorded in INEX's use of XPath and NEXI at the 2003 and 2004 Workshops were 63% and 12% of proposed queries were either syntactically or semantically incorrect [7]. Therefore, if experts in the field of structured information retrieval are unable to correctly use complex query languages, one cannot expect a casual user to do so. This theory was verified by researchers in INEX's interactive track [14] who observed the difficulty that casual users had in formatting formal queries. However, we feel that users would be able to intuitively express their information need in a natural language.

A second problem with formal query languages is that they are too tightly bound to the physical structure of documents; and therefore, users require an intimate knowledge of the documents' composition in order to fully express their structural requirements. So, in order for users to retrieve information from abstracts, bodies or bibliographies, they will need to know the actual names of those tags in a collection (for instance: *abs*, *bdy*, and *bib*). While this information may be obtained from a document's DTD or Schema there are situations where the proprietor of the collection does not wish users to have access to those files. Or, in the case of a heterogeneous collection, a single tag can have multiple names (for example: abstract could be named *abs*, *a*, or *abstract*). This is a problem identified by participants in the INEX 2004 heterogenous track who have proposed the use of metatags to map between collections [6] and extensions to NEXI [10] to handle multiple tag names. Naturally, neither of these solutions are trivial, which is why INEX has multiple tracks (heterogenous and document mining) devoted to investigating this problem. In contrast, structural requirements in NLQs are inherently expressed at a higher conceptual level, allowing the underlying document's structure to be completely hidden from users, although NEXI could also be extended to handle conceptual tag names.

3 Previous Work by Authors

This paper expands on the previous work of the authors [12,13]. We submitted our system, NLPX, to INEX's the 2004 and 2005 Natural Language Processing track where it has performed strongly. INEX's NLP track used the same topics and assessments as its Ad-hoc track; however, participating systems used a natural language query as input, rather than a formal language (NEXI) query. Examples of both query types are expressed in Figure 1. Note that the query actually contains two information requests, first, for sections about compression, and second, for articles about information retrieval. However, the user only wants to receive results matching the first request. We refer to the former as returned requests/results and the latter as support requests/results.

NEXI: //article[about(.,'information retrieval')] //sec[about(./, compression)]

NLQ: Find sections of articles about image and text compression in articles about efficient information retrieval

Fig. 1. A NEXI and Natural Language Query

As with last year, the goal of the 2006 NLP participants was to produce a natural language interface that translated NLQs to into NEXI queries. The translated queries were executed by a single backend system (GPX) and the retrieval performance of translated queries was recorded as if it were a standard Ad-hoc system. Participants in the NLP track compare their retrieval performance amongst each other as well as a baseline system that uses manually formed NEXI expressions (that is the original *title* tags) as input. NLPX's translation process involves four steps that derived syntactic and semantic information from the natural language query (NLQ). We refer to these four steps as the NLPX framework and outline them below:

1. First, NLPX tags words in the NLQ as either a special connotation or by their part of speech. Special connotations are words of implied semantic significance. Words corresponding to special connotations can either be hard-coded into the system and matched to query words by a dictionary lookup or tagged using a modified Brill Tagger that also considers a word's context when tagging. Non-connotations are tagged by their part of speech (such as noun, verb, conjunction) via a Brill Tagger [2].
2. Second, words are grouped together into phrases using a process called Chunking [1]. The reason that NLPX recognises Chunks is to reduce ambiguity and to facilitate further content analysis at a later stage. There are three main types of chunks NLPX recognises: Instructions (for example "I want to retrieve"), Structures (for example "sections of articles") and Content (for example "information retrieval").
3. Third, NLPX matches the tagged NLQs to query templates. The templates were derived from the inspection of previous INEX queries. Since the NLQs occurred in shallow context they required only a few templates, significantly less than if one wished to capture natural language as a whole. Each template corresponded to an information request. Each request had three attributes: Content, a list of terms/phrases expressing content requirements, Structure, a logical XPath expression expressing structural requirements, and an Instruction, "R" for return requests, and "S" otherwise.
4. Finally, the requests are merged together and output in NEXI format. Return requests are output in the form **A[about(.,C)]** where **A** is the request's structural attribute and **C** is the request's content attribute. When all return requests are processed, support requests were inserted. The insert position was located by comparing the structural attributes of return and support requests and by finding their longest shared descendant. The output of support requests had the form **D[about(E,F)]** where **D** is the longest matching string, **E** is the remainder of the support's structural attribute and **F** is the support's content attribute. Note, that while NLPX outputs NEXI queries this step has been modulated so that NLPX could be extended to include any number of formal query languages.

4 Improvements

As usual, we have made several improvements to NLPX since last year's participation. However, the number of improvements was less than in previous years that may indicate the research is reaching a plateau. The major change this year was porting NLPX to the new Wikipedia collection. Fortunately, this was not an extraneous task since the only change that was required was in the third step. Other changes made this year was the addition of strengtheners, that is content words signalled by the user as being of high importance, and a constraint added by the system to ensure that support requests are not placed in the last about clause even if they have the same return type as return requests (as defined by NEXI standards).

5 System Backend

Once the NLQ was tagged, chunked and matched to templates it was transformed into a NEXI query using the existing NLPX system. This is a two stage process. First we expanded the content of the query, by deriving phrases based on its lexical properties, such as noun phrases that include adjectives and participles. Then we formatted a NEXI query based upon its instruction, structure and content values. We passed the NEXI query to our existing GPX system for processing as if it were a standard Ad-hoc query. To produce its results list GPX collects leaf elements from its index and dynamically creates their ancestors. GPX's ranking scheme rewards leaf elements with specific rather than common terms, and elements that contain phrases. It also rewards ancestors with multiple relevant children rather than a single relevant child. A more comprehensive description of GPX can be found in our accompanying paper as well as earlier work [5].

6 Results

Here we present the performance results from NLPX. The results are split into two parts. The first part discusses how well NLPX was able to translate natural language queries to NEXI. The second part presents the retrieval performance of the translated queries in comparison with the original NEXI queries.

6.1 Translation Performance

The *description* elements of the 125 INEX Ad-hoc topics were translated in NEXI format by NLPX. These translations were then manually compared with each topic's *description* and *castitle* elements to test their accuracy. Furthermore, a comparison was made between each *castitle* and *description* elements themselves, to test how faithfully the topic's originator was able to express their information need in natural and formal language. The results of these comparisons are presented in Table 1.

Table 1. Comparison between translation, description and castitle

	Description	**Castitle**
Translation	0.704	0.352
Description		0.408

The overall accuracy rate between the translation and *descriptions* was high (70.4 per cent), however the similarity between the translations and the original *castitles* was much lower (35.2 percent). This is largely because the similarity between the *descriptions* and *castitles* were also low (40.8 percent) Examples of successful translations were seen in INEX topics 292 and 311 as shown in Figures 2 and 3.

Description: I want figures containing Renaissance paintings of Flemish or Italian artists, but not French or German ones
Translation: //article//section//(figure|caption)[about(.,"Renaissance paintings" Renaissance paintings Flemish "Italian artists" Italian artists "-German -ones" -German -ones)]
Title: //article//figure[about(., Renaissance painting Italian Flemish -French -German)]

Fig 2. INEX Topic 292

Description: Find sections about global warming cause and effects.
Translation: //article//section[about(.,"global warming cause" global warming cause "global cause" "warming cause" effects)]
Title: //section[about(., global warming cause and effects)]

Fig 3. INEX Topic 311

However, other times NLPX was unsuccessful in translating the *description*. Examples of this occurred in INEX topics 310 and 358, shown in Figures 4 and 5. For Topic 310 NLPX was unable to determine that a second information request regarding sections was made by the user, while for Topic 358 NLPX was unable to recognise the term “information” as content-bearing rather than as a structural request.

Description: Find articles about Novikov self-consistency principle that contain a section about time travel.
Translation: //article[about(.,"Novikov self consistency principle" "Novikov self-consistency principle" "self consistency" self consistency Novikov self-consistency principle "Novikov principle") OR about(.,"time travel" time travel)]
Title: //article[about(., Novikov self-consistency principle) and about(.//section, time travel)]

Fig 4. INEX Topic 310

Description: Retrieve sections of articles about the use of ontologies in information retrieval such as semantic indexing.
Translation: //article[about(.//*,retrieval "semantic indexing" semantic indexing)]//section[about(.,ontologies)]
Title: //article//section[about(.,ontologies information retrieval semantic indexing)]

Fig 5. INEX Topic 358

Furthermore, a lot of times the *castitle* (59.2) and *description* did not match, particularly in terms of structural requests. This is worrying since the *castitle* and *description* should be faithful representations of the user's information need. For instance, in INEX Topic 402, shown in Figure 6, the *description* asks for ***information*** about capitals of European counties, presumably this means that the structural constraints are arbitrary and therefore a wildcard element (//*) should be specified as the NEXI path (essentially turning the topic into a Content Only query). However, the *title* specially specifies ***articles*** as the NEXI path. Another example occurs in INEX topic 400, shown in Figure 7, where the *description* specifically requests ***documents*** but the *title* specifies for ***sections*** in the NEXI path. Therefore, in these types of instances it would be impossible for the translated query to match both the *description* and *title*.

Description: Find information on capitals of European countries.
Translation: //article//*[about(.,capitals "European countries" European countries)]
Title: //article[about(.,country european)]//section[about(.,capital)]

Fig 6. INEX Topic 402

Description: Find documents about countries that had non-violent revolutions
Translation: //article[about(.,countries "non violent revolutions" "non-violent revolutions" "non violent" non violent non-violent revolutions)]
Title: //article[about(., country revolutions)]//section[about(., "non violent")]

Fig 7. INEX Topic 400

6.2 Retrieval Performance

As with previous years the translated queries from NLPX and the other NLP participants were executed by a single backend system GPX. This allowed for the same analysis of the NLP participants retrieval performance as is seen in traditional information retrieval. An additional "baseline" consisting of the original *castile* queries was also executed by GPX, allowing for a comparison to be made between automatic and manually created NEXI statements. Furthermore, since the same INEX topics and relevance assessments are used by the NLP and Ad-hoc tracks cross-track comparisons are valid. At the time of publication results for the Thorough, Focussed and Best-In-Context tasks were available. This section begins with a detailed analysis of NLPX's performance in the Thorough task before presenting an outline of NLPX's performance across all tasks. Also, note that the set of relevance judgements used to evaluate these systems included all relevant elements, including those removed form later relevance assessments for being too small (mainly link elements).

As with previous years, the aim of INEX's Thorough task was to retrieve as many relevant information items as possible, regardless of how many "overlapping" items were returned. The metric used to evaluate the Thorough task is Mean Average effort-precision (MAep), analogous to traditional mean average precision. NLPX produced two translations for each of the tasks: one produced by the current implementation of NLPX (NLPX06) and one produced by a previous implementation of NLPX (NLPX05). GPX produced two submissions for each of the translations , one that one that favoured leaf elements and one that favoured root elements. The MAep results of each of the submissions are presented in Table 2. The results show that NLPX performed very strongly, outperforming alternative NLP approaches. Furthermore, it performed comparable to the Baseline achieving a ratio of around 0.8. This is highlighted in the effort-precision gain-recall graph presented in Figure 8 where the NLPX and Baseline submissions produce similar plots.

Table 2. MAep Results of the 2006 INEX Thorough Task

Translation	Root Orientated	Leaf Orientated
Baseline (NEXI)	0.0323	0.0284
NLPX06	0.0298	0.0251
NLPX05	0.0279	0.0229
Robert Gordon	0.0235	0.0171
Robert Gordon 2	0.0224	0.0159
Exoles des Mines de Saint-Erienne	0.0235	0.0205
Exoles des Mines de Saint-Erienne XOR	0.0231	0.0197

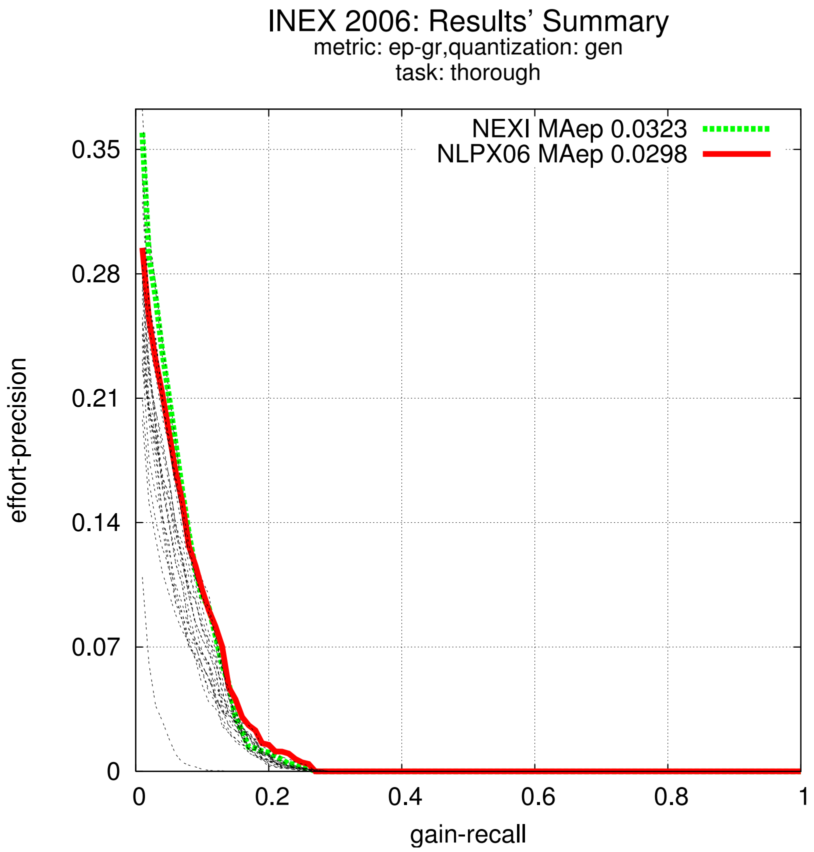

Fig 8. INEX 2006 NLP Through ep-gr graph

Similar results were achieved by NLPX across all the tasks. Table 3 shows the retrieval performance of the best performing NLPX, Baseline and Ad-hoc submission for the Thorough, Focussed and Best-in-Context tasks. The table contains both the retrieval score (using the appropriate metric) and the pseudo-rank of the submission if it was submitted to the Ad-hoc track. Once again, NLPX performs strongly in the Thorough task, and while its rank is affected severely in the other two tasks, its ratio to the Baseline remains fairly consistent. It must be noticed however, that these results are derived from the original INEX pools that rewarded link elements very highly. However, GPX removed all links from each of the submissions (including all NLP participants). Hence, different scores would be recorded if the second INEX pool was used that scored link elements as too small to be relevant. In this scenario, the retrieval scores and pseudo-ranks of the Baseline and NLPX submission would probably increase, however the ratio between the scores should not be greatly affected.

Table 3. Retrieval performance across all tasks

Task Metric	Thorough Maep	Focussed Overlap On nxCG[50]	Focussed Overlap Off nxCG[50]	Best-In-Context BEPD At A=100.0	Best-In-Context EPRUM At A=100.0
Best NLPX	0.0298 (15)	0.139 (47)	0.1989 (21)	0.6551 (37)	0.1974 (44)
NEXI Baseline	0.033 (10)	0.173 (20)	0.2405 (10)	0.6895 (26)	0.2243 (33)
Best Adhoc	0.0384 (1)	0.2265 (1)	0.2802 (1)	0.7983 (1)	0.3146 (1)

7 Conclusion

This paper presented the results of NLPX's participation in INEX 2006. This is the third year that NLPX has participated in INEX's NLP task, each year improving its performance. This year it was able to correctly translate a majority of topics from natural language to formal language. It outperformed the alternative NLP approaches and was comparable to a baseline formal language system. These results validate the potential of natural language queries as alternative to formal language queries in the domain of XML retrieval.

References

1. Abney, S.: Parsing by Chunks. In: Principle-Based Parsing. Kluwer Academic Publisher (1991)
2. Brill, E.: A Simple Rule-Based Part of Speech Tagger. In: Proceedings of the Third Conference on Applied Computational Linguistics (ACL), Trento, Italy (1992) 152–155
3. Clark J., DeRose, S.: XML Path Language XPath Version 1.0. W3C Recommendation, The World Wide Web Consortium, November 1999 available at http://www.w3.org/TR/xpath.
4. Fox, C: Lexical Analysis and Stoplists. In: Frankes, W.B., Baeza-Yates, R. (eds.): Information Retrieval: Data Structures and Algorithms, Prentice-Hall, Upper Saddle River, New Jersey, United States of America (1992) Chapter 7 102-130.
5. Geva, S.: GPX - Gardens Point XML Information Retrieval INEX 2004. In: Fuhr, N., Lalmas, M., Malik, S., Szlavik Z. (eds.): Advances in XML Information Retrieval: Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl, Germany, December 6–8, 2004, Revised Selected Papers. Lecture Nodes in Computer Science, Vol 3493. Springer-Verlag, Berlin Heidelberg New York (2005) 221–222

6. Larson, R.: XML Element Retrieval and Heterogenous Retrieval: In Pursuit of the Impossible? In Proceedings of INEX 2005 Workshop on Element Retrieval Methodology, Glasgow, Scotland (2005) 38-41.
7. O'Keefe, R., Trotman, A.: The Simplest Query Language That Could Possibly Work, In: Fuhr N., Malik, S. (eds.): INEX 2003 Workshop Proceedings. Dagstuhl, Germany (2003) 167–174
8. Ramshaw, L. Marcus, M.: Text Chunking Using Transformation-Based Learning, In: Proceedings of the Third Workshop on Very Large Corpora (1995) 82-94.
9. Trotman, A., Lamas, M.: Why structural hints in queries do not help XML-retrieval, In: Efthimiadis E. N., Dumais S. T., Hawking, D., Järvelin, K. (eds.):, SIGIR 2006: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, August 6-11, (2006) 711–712
10. Trotman, A., Sigurbjörnsson, B.: NEXI: Now and Next, In: Fuhr, N., Lalmas, M., Malik, S., Szlavik Z. (eds.): Advances in XML Information Retrieval: Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl, Germany, December 6–8, 2004, Revised Selected Papers. Lecture Nodes in Computer Science, Vol 3493. Springer-Verlag, Berlin Heidelberg New York (2005) 410–423
11. Woodley, A., Geva, S.: NLPX: An XML-IR System with a Natural Language Interface, In: Bruza, P., Moffat, A., Turpin, A (eds.): Proceedings of the Australasian Document Computing Symposium, Melbourne, Australia (2004) 71–74.
12. Woodley, A., Geva, S.: NLPX at INEX 2004, In: Fuhr, N., Lalmas, M., Malik, S., Szlavik Z. (eds.): Advances in XML Information Retrieval: Third International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2004, Dagstuhl, Germany, December 6–8, 2004, Revised Selected Papers. Lecture Nodes in Computer Science, Vol 3493. Springer-Verlag, Berlin Heidelberg New York (2005) 393–406
13. Woodley, A., Geva, S.: NLPX at INEX 2005, In: Fuhr, N., Lalmas, M., Malik, S., Szlavik Z. (eds.): Advances in XML Information Retrieval: Fourth International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl, Germany, November 28 – 30, 2005, Revised Selected Papers. Lecture Nodes in Computer Science, Vol 3977. Springer-Verlag, Berlin Heidelberg New York (2005) 358–372
14. van Zowl, R., Bass, J., van Oostendorp, H., Wiering, F.: Query Formulation for XML Retrieval with Bricks. In Fuhr, N., Lamas, M., Trotman, A. (eds.): In Proceedings of INEX 2005 Workshop on Element Retrieval Methodology, Glasgow, Scotland (2005) 75-83.