A Multi Stage Fall-back Search Strategy for Cross-Lingual Information Retrieval

Satish Kagathara, Manish Deolalkar, Pushpak Bhattacharyya
Media Lab Asia, KReSIT, IIT Bombay
Mumbai – 400076 India
{satishk, manishpd}@it.iitb.ac.in, pb@cse.iitb.ac.in

Abstract

In this paper, we describe a special purpose search engine in the Agricultural domain. The engine is called *AgroExplorer,* which is designed to search and retrieve the contextual information relevant to the users in their own languages, preferably. In order to facilitate this functionality, the system extracts the *meaning* of a query. The *meaning* is represented in the form of Universal Networking Language (UNL) expressions. The system performs the search using UNL *expression* matching. In case the *complete expression* matching does not succeed, the method is backed up by (a) *partial expression* matching (b) *concept (Universal Word (UW))* matching and (c) traditional *keyword-based* matching. This endows the search engine with *multilingual* capability and higher precision due to the use of semantics. The languages of focus currently are English, Hindi and Marathi. The design of the system, however, does not make any assumption about the languages involved, as long as there are analyzers and generators of UNL expressions.

Keywords

Cross-Lingual Information Retrieval, *AgroExplorer*, Universal Networking Language (UNL), UNL expression, UNL graph, Expression Matching, Universal Word Matching, Keyword Matching.

1 Introduction

Tracking down relevant information quickly on the Internet remains an elusive quest. Because the Web is not indexed in any standard manner, finding information can be difficult. Search engines are popular tools for locating web pages, but they often return thousands of results. Search engines crawl the Web and log the words from the web pages they find in their databases. Because some search engines have logged the words from over billions of documents, results can be overwhelming. Without a clear search strategy [7], using a search engine is like wandering aimlessly in the stacks of a library trying to find a particular book.

To use search engines effectively, it is essential to apply techniques that narrow results and push the most relevant pages to the top of the results list. Today, most of the search engines (Google, AltaVista, Yahoo *etc.*) are *keyword-based*. They follow different search strategies like *Boolean search, Phrase search etc.* on keywords. But this is hardly sufficient to serve the user need in the context. In this paper we describe a *multi stage fall- back search strategy* that exploits the semantics of the queries and the information base. This leads to a *meaning-based, multilingual* capability of the system.

2 Related Work

This section gives a brief review of work related to this project. This allows us to put our model in perspective.

2.1 Existing Meaning-based Search

Only a few *meaning-based* search engines have been developed so far. Search engines like *oingo.com*, *excite.com* and *simpli.com* [10] also provide *meaning-based* search. Launched in October 1999, *Oingo* has already introduced three fully functional products: *DirectSearch*, *DomainSense* and *AdSense*. *DirectSearch*, a *meaning-based* search technology, uses the company's ontology to provide more precise and effective search results. *DomainSense*, *Oingo*'s *meaning-based* domain name suggestion technology, currently increases domain name sales for leading registrars around the world. *AdSense* serves the most highly targeted advertisements on the Internet, effectively targeting advertisements based on search meanings rather than keywords. The attempts made are highly laudable but their results are nowhere close to the mark.

Clush [11] is a new engine, which produces clustered search results from millions of web pages giving the user dynamically categorized data that cannot be duplicated by static or human edited directories. It looks like the old "*concept*" or *meaning-based* searches for the late 90s, similar to *Oingo* and *Simpli*. Likely Meaning Score (LMS), combined with clustering, allows *Clush* to truly understand the *meaning* of query. For example, for the query *"virus"*, *Clush*'s LMS (Likely Meaning Score) yields results on biological viruses, as the likely meaning of the single word query *virus* has a higher probability of being related to biology rather than computer *anti-virus*. This engine works well with one word or at most two. But the relevancy of the actual web results is disappointing.

2.2 Existing Multilingual Information Retrieval System

Kazhugu, a *multilingual* Internet search engine, claimed to be India's first in regional languages, has been developed by Anna University-KB Chandrasekhar (AU-KBC) Research Centre, Chennai. Tentatively named *'Kazhugu'* (Tamil for

eagle), the search engine is ready for use in Tamil websites (one can do both site specific and web searches), The Research Centre will soon come out with similar Internet search engines for Hindi, Malayalam, Telegu and Kannada languages. The search engine will be placed for testing on the Internet portal of Sify Ltd.

Another Step in the direction of Information Retrieval was taken by *MIETTA* [6] (Multilingual Information Extraction for Tourism and Travel Assistance) project whose objective is to develop a cross-lingual information management platform that can be marketed to industry. The system resulting from the *MIETTA* project will allow retrieval of tourist information in several languages (English, Finnish, French, German, Italian) and on a number of different geographical regions (the German federal state of Saarland, the Southwestern Finnish region centered around Turku and the Italian city of Rome).

3 Our System: *AgroExplorer*

AgroExplorer [1] [1] a *multilingual, meaning-based* search engine. "Fig. 1," shows the overview of the system architecture with the important modules. The system flow is described below.

The Web is crawled [2] for agricultural documents using the Focused Crawler, and the HTML Corpus is built. The sentences generated by the HTML Parser are given to the Enconverter [4] system that converts them to the UNL form. The Indexer module builds indices [2] on the UNL expressions after preprocessing it.

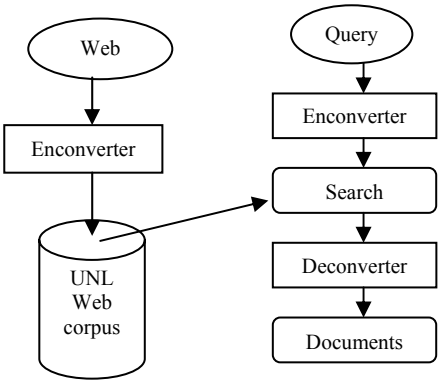

Fig. 1. System Architecture.

Once the user enters a query, we first get the UNL of the query by passing it through the Enconverter. The Search module takes the UNL expression of the query and performs a *meaning-based* search on an *inverted index* created earlier, using the following *fall-back* search strategies:

a. *Complete expression matching*
b. *Partial expression matching (if a. fails)*
c. *Universal Word (UW) matching (if a. and b. fail)*
d. *Keyword matching (if a., b. and c. fail)*

[1] http://agro.mlasia.iitb.ac.in

The Search module returns the UNL documents which are passed to the Deconverter [5] (*i.e.,* generator) system, which converts them into the target language. Finally, the system displays the translated documents to the user.

The system is using *Jakarta Lucene* [8], an open source Unicode compliant search tool for performing a keyword search. The query is also given to the *Google* search engine [9].

4 Universal Networking Language (UNL)

The Universal Networking Language (UNL) [3] is an electronic language for computers to express and exchange information. The UNL consists of Universal Words (UWs), Relations, Attributes, and the UNL Knowledge Base. The Universal Words constitute the vocabulary of the UNL, Relations and Attribute constitutes the syntax of the UNL, and the UNL Knowledge Base (KB) constitutes the semantics of the UNL. The KB defines possible relationships between UWs.

4.1 Binary Relations

Binary Relations are the building blocks of UNL documents. They are made up of a relation and two UWs.

<Binary Relation> = <Relation Label> [":"<Compound UW-ID>] "({ <UW_1> [":" <$UW\text{-}ID_1$>]} | { ":" <Compound $UW\text{-}ID_1$>}[<Attribute List>] ","{ <UW_2> [":" <$UW\text{-}ID_2$>]} | { ":" <Compound $UW\text{-}ID_2$>}[<Attribute List>] ")"

Where,
Relation Label: String of three lowercase characters. *e.g.* agt
Compound UW-ID: Two digits string identifying instance specified by Compound UWs.
UW-ID: The UW-ID is used to indicate some referential information, for example that there are two or more different occurrences of the same concept (they are not co-referent).
Attribute: Provides information of how concept is being used. *e.g.* @past, @plural
Universal Word (concept): A UW is made up of a character string (an English language word) followed by a list of constraints.
<UW> = <Head Word> [<Constraint list>]
Where,
Head Word: An English word/compound word/phrase/sentence that is interpreted as a label for a set of concepts that may correspond to that in English.
Constraint: Restricts the interpretation of a UW to a subset or to a specific concept.

4.2 UNL Example

UNL represents information sentence-by-sentence as a hyper-graph with concepts as nodes and relations as arcs as shown in "Fig. 2," below for the sentence *"John eats rice with a spoon"*.
UNL expression:
agt(eat(icl>do).@entry.@present, John(iof>person))
obj(eat(icl>do).@entry.@present, rice(icl>food))
ins(eat(icl>do).@entry.@present, spoon(icl>artifact))

UNL graph:

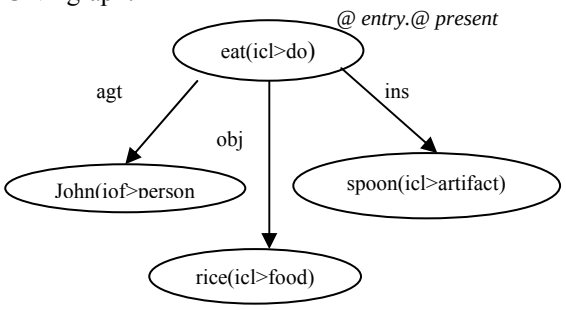

Fig. 2. UNL graph of "*John eats rice with a spoon*".

In above figure, the arcs labeled with *agt* (agent), *obj* (object) and *ins* (instrument) are the relation labels. The nodes *eat (icl>do); John (iof >person), rice (icl>food)* and *spoon (icl>artifact)* are the *Universal Words* (*UW*). These are language words with *restrictions* in parentheses. *icl* stands for *inclusion* and *iof* stands for *instance of*. UWs can be annotated with attributes like *number*, *tense etc.*, which provide further information about how the concept is being used in the specific sentence. Any of the three restriction labels - *icl, iof* and *equ* can be attached to an UW for restricting its sense.

4.3 UNL Document

Any component, such as a word, phrase or title and, of course, a sentence of a natural language can be represented with UNL expressions. A UNL expression therefore consists of a UW or a (set of) binary relation(s). In UNL documents, a UNL expression for a sentence is enclosed by the tags {unl} and {/unl} inside [S] and [/S]. If a UNL expression consists of a UW, this UW should be enclosed further by the tags [W] and [/W].

UNL expression:

{unl}
<Binary Relation>
...
{/unl}

5 Indexing

The effectiveness of any search engine mainly depends on its index structure. Its structure should be capable of catering sufficient and relevant information to the users in terms of their needs. To serve users with the contextual information in their own language, the system needs to index on *meaning* representation (UNL expressions) and not just on *plain text*. Also the size of the index should not be too large.

In our case, The UNL expressions contain some extraneous information, which is not needed by the system. So, after preprocessing it, the Indexer generates an *inverted index* on the UNL documents. It parses the UNL expressions and extracts the relation, the two UW's on which the relation has been formed (UW1 and UW2) and their respective UW-IDs. The UNL index is stored in a database and it consists of *UNL document index* and *UNL index*.

The *UNL document index* (see Table 1) keeps information about each UNL document. The information stored in each entry includes the document id, link and language of the original document and number of sentences in it.

Table 1
UNL Document Index

Fields	Description
docid	UNL document id
orilink	Link to original document
language	Language of original document
numlines	Number of sentences in the document

The *UNL index* (see Table 2) stores the actual index of UNL expressions. Each entry keeps the information of UNL relation, its UW1 and UW2 and their respective UW-IDs and the sentence number and its document id where the combination of relation, UW1 and UW2 occurs.

Table 2
UNL Index

Fields	Description
rel	Stores the relation of a UNL expression
uw1	Stores the first Universal Word of a relation
uw2	Stores the second Universal Word of a relation
uwid1	Stores the id of uw1
uwid2	Stores the id of uw2
docid	UNL document id in which above fields occur
sent	Sentence no in which above fields occur

Statistics of Index:

Table 3 shows the statistics of UNL index.

Table 3
Statistics of Index

Total number of indexed document	39
Total distinct UNL expression	around 3468
Total distinct UWs	around 3478

The following example illustrates the index representation of the UNL expression of one sample sentence.

docid: agro4
sentence:
[S:1]
Institute of Development Studies at Jaipur has taken up a comprehensive all-India research project 'Equity-driven trade and marketing policy strategies for improved performance of Indian Agriculture' on agricultural reforms with financial support from the government.
[/S]
UNL expression:
{unl}
…
obj:02(with:4A.@entry, support(icl>help):4T)
mod:02(support(icl>help):4T, :01)
mod:02(support(icl>help):4T, financial(mod<thing):4J)
obj:01(from:51.@entry, government(icl>governmental organization):5L.@def)
mod:01(government(icl>governmental organization):5L.@def, australian(mod<thing):5A)
pur:03(:04.@entry, :05)

and:04(strategy(icl>idea):1D.@entry.@pl,
trade(icl>activity):0I)
mod:04(strategy(icl>idea):1D.@entry.@pl,
policy(icl>plan):16)
mod:04(policy(icl>plan):16, marketing(icl>commerce):0W)
mod:04(trade(icl>activity):0I, equity-driven(mod<thing):0A)
obj:05(improve(icl>change(agt>thing,obj>thing)):22.@past,
performance(icl>operation):2B.@entry.@topic)
mod:05(performance(icl>operation):2B.@entry.@topic,
agriculture(icl>activity):2X)
mod:05(agriculture(icl>activity):2X, indian(aoj>thing):2Q)
{/unl}

UNL index:

The UNL index is shown in "Fig. 3." Each entry *(UNL expression (rel, uw1, uw2))* points to the pair of document id and its sentence number where it occurs.

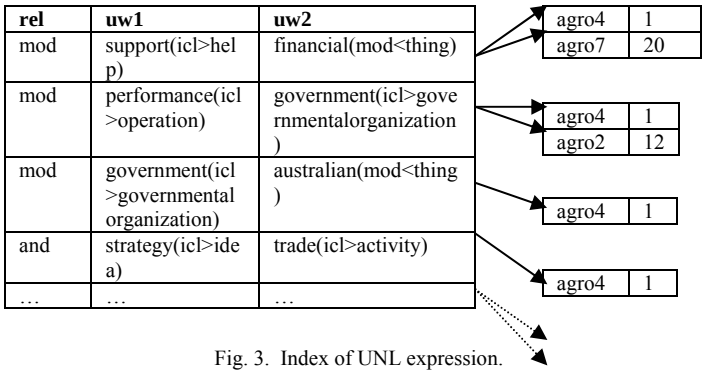

Fig. 3. Index of UNL expression.

"Fig. 4," shows that each Universal Word points to the pair of document id and its sentences number where it occurs.

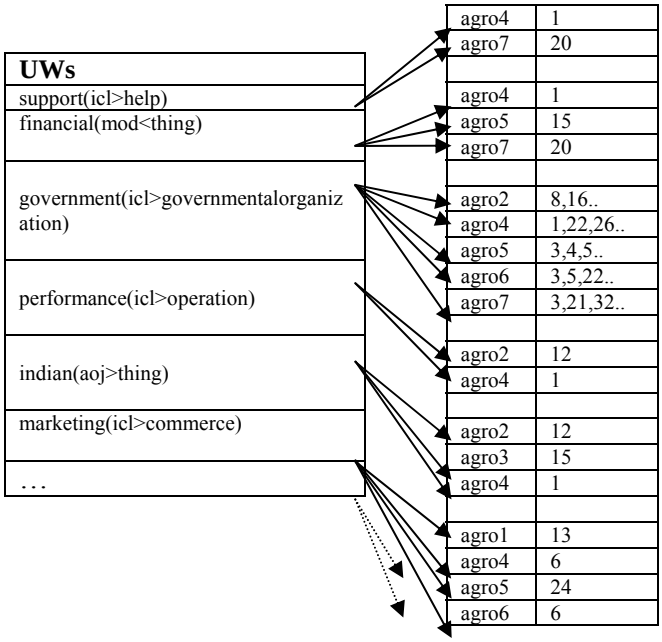

Fig. 4. Index of UWs.

6 Fall-back Search Strategy

Users have different preferences about the type of information they typically require and the ways in which the information is searched. There are a number of factors that affect the responsiveness of a system to such requests. An obvious example of this is a changing emphasis on recall versus precision, the two most important measures used for judging the results of a search engine. Such cases indicate the potential usefulness of alternative search strategies. In the following section we discuss the search strategies in more detail as we briefly presented them above, concentrating on relevance and ranking [2].

6.1 Complete Expression Matching

The relevance of a document depends on the query posed by the user. As explained earlier, both the query and the document are converted into UNL before a search is performed. For each sentence in the document, we have one UNL graph. Thus, essentially, we have a collection of UNL graphs of document and a given UNL graph of query, which needs to match with the document. If the query UNL graph is a subgraph of any sentence UNL graph in the document, then we can say that the document containing that sentence is completely relevant to the query and the search engine should retrieve it. We need to do a subgraph matching between the query UNL graph and the sentences UNL graph. Mathematically, this can be expressed as,

$$R_q(d) = \frac{\Sigma_{s \in Sd} r_q(s)}{|S_d|} \quad (1)$$

Where,

$R_q(d)$=Relevance of the document d to the query q
S_d=Number of sentences in the document d
$r_q(s)$=Relevance of sentence s to the query q
$r_q(s)$= 1 if the query UNL graph is a subgraph of the sentence UNL graph,
0 otherwise

But, the drawback of this is that it is a one-or-none matching approach, *i.e.* a sentence will be either relevant to a query or it will not be. There is no concept of *partial expression* matching. Undoubtedly, *complete expression* matching approach will lead to high precision but low recall. To overcome this problem, we can introduce a *partial expression* matching scheme which has lower precision but higher recall.

6.2 Partial Expression Matching

Here, it is not necessary for a query UNL graph to be a subgraph of sentence UNL graph. It can be a part of it. Thus, $r_q(s)$, the relevance of a sentence *s* to the query is not 1.0 here. Also it is often the case that the two UWs (vertices) are two different instances of the same concept, which is indicated by their UW-IDs. So, the two different occurrences of the same UW in two different relations (edges) of the query are said to be linked if the UW-IDs of these two occurrences are same and

this link is said to be the *correct* link if it is also present in the sentence UNL graph.

To find the relevance of a sentence *s* to the query, we need to find the number of times the relations of the query UNL graph are found in the sentence UNL graph and the number of *correct* links between them. The relevance of a document $R_q(d)$ is the same as "(1)," that is given in the *complete expression* matching approach. For *partial expression* matching the $r_q(s)$ can be expressed as,

$$r_q(s) = \alpha \frac{n}{N} + (1-\alpha)\frac{l}{L} \quad (2)$$

Where,

n=Number of times the relations (of the query UNL graph) found in the sentence UNL graph
N=Total number of relations in the query UNL graph
l=Number of *correct* links between UNL graph of query and sentence
L=Total number of links between the query UWs
α =Empirical constant

It is possible that there is no common relation between the UNL graph of query and that of the sentence. This strategy fails in this case. But, the *Universal Word* matching approach which has lower precision but higher recall compared to *partial expression* matching solves this problem.

6.3 Universal Word Matching

As explained earlier, Universal Word is a Head Word with a constraint list. It is not just a keyword, but it also disambiguates the sense of word. Hence, this technique gives higher precision but lower recall compare to *keyword-based* matching.

To find $R_q(d)$, the relevance of the document d to the query q, for this case, it requires the total number of occurrences of query UWs in the UNL document and the highest number of occurrences of query UWs in any UNL document. It can be expressed as,

$$R_q(d) = \frac{\Sigma_{s \in Sd}\, r_q(s)}{\text{MaxScore}} \quad (3)$$

Where,

$r_q(s)$=Total number of occurrences of query UWs in the UNL expression of sentence s
MaxScore= Highest number of occurrences of query UWs in any UNL document

If search fails in case of *complete expression* matching, it automatically falls back on *partial expression* matching - the second stage. Finally, in case of failure of *partial expression* matching it performs the search on Universal Words. Also the system facilitates the user to view the search results of each stage separately.

7 Results

We show the results of the system for some sample queries to demonstrate how the system performs search at each stage. The examples are, for the 6 cases of

a. *Complete expression matching*
b. *Partial expression matching*
c. *Universal Word matching*
d. *Keyword-based matching*
e. *Meaning-based search*
f. *Multilingual search*

The information base of the search engine consists of a total number of 39 indexed agricultural documents.

a. Complete Expression Matching

Sample Query: *financial support from the government*
UNL of the Query:
mod(support(icl>help):0A.@entry, :01)
mod(support(icl>help):0A.@entry, financial(mod<thing):00)
obj:01(from:0I.@entry,government(icl>governmental organization):0R.@def)
UNL Graph:

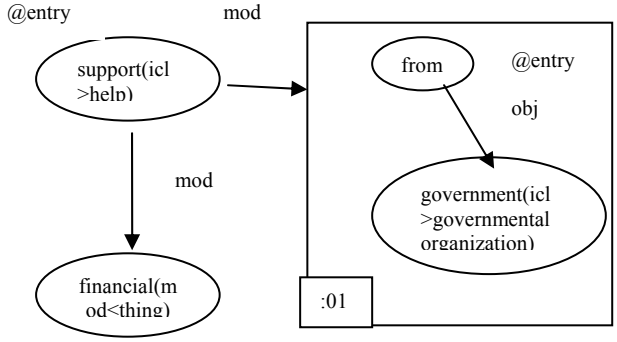

Fig. 5. UNL graph of "*financial support from the government.*"

Matched UNL documents and relevant sentences therein:
Total match found: 1
docid: agro4
relevance: 1
[S:1]
Institute of Development Studies at Jaipur has taken up a comprehensive all-India research project 'Equity-driven trade and marketing policy strategies for improved performance of Indian Agriculture' on agricultural reforms with financial support from the government.
{unl}
…
obj:02(with:4A.@entry, support(icl>help):4T)
complete expression match
mod:02(support(icl>help):4T, :01)
mod:02(support(icl>help):4T, financial(mod<thing):4J)
obj:01(from:51.@entry, government(icl>governmental organization):5L.@def)

mod:01(government(icl>governmental organization):5L.@def, australian(mod<thing):5A)
…
{/unl}
[/S]

In the above, the dotted ellipses represent the expressions in the sentence, which match the query, *i.e., the query UNL graph is a subgraph of the sentence UNL graph.*

b. Partial Expression Matching

Sample Query: *essential tool for farm management*
UNL of the Query:
mod(tool(icl>functional thing):0A.@entry, :01)
mod(tool(icl>functionalthing):0A.@entry, essential(aoj>thing):00)
obj:01(for:0F.@entry, farm management(icl>activity):0J)

UNL Graph:

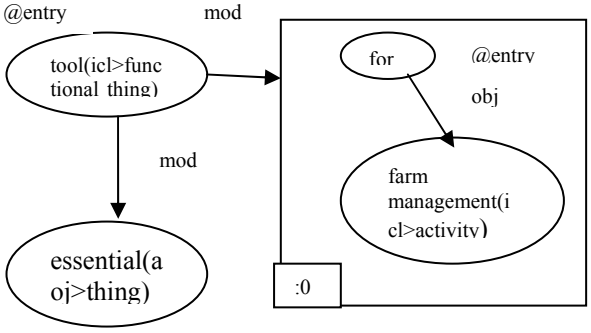

Fig. 6. UNL graph of "*essential tool for farm management.*"

Matched UNL documents and relevant sentences therein:
Total match found: 1
docid: agro1
relevance: 1
[S:22]
Computers have become an essential tool for farm management.
{unl}
…
obj(become(icl>transform{>change}(obj>thing,gol>thing,src>thing)):0F.@entry.@present.@complete.@pred, tool(icl>functional thing):0Z.@indef)
partial expression match
mod(tool(icl>functional thing):0Z.@indef, essential(aoj>thing):0P)

mod(management(icl>activity):1D, farm(icl>place):18)
…
{unl}
[/S]

In the above, the dotted ellipses represent the expressions in the sentence, which partially match the query.

c. Universal Word Matching

Sample Query: *marketing of fruits and vegetables*
UNL of the Query:
mod(marketing(icl>commerce):00.@entry, :01)
and:01(vegetable(icl>food>functional thing,icl>plant>living thing):0O.@entry.@pl,fruit(pof>plant):0D.@pl)

Matched UNL documents and relevant sentences therein:
Total match found: 8 (shown three here)
docid: agro5
relevance: 1
[S:5]
…
aoj(promote(icl>support(agt>thing,obj>thing)):0M.@entry.@present.@progress,government(icl>governmental organization):04.@def)
plc(promote(icl>support(agt>thing,obj>thing)):0M.@entry.@present.@progress, country(icl>region):2K.@def)
obj(promote(icl>support(agt>thing,obj>thing)):0M.@entry.@present.@progress, *marketing(icl>commerce)*:16)
mod(*marketing(icl>commerce)*:16, organized(aoj>thing):0W)
mod(*marketing(icl>commerce)*:16, commodity(icl>goods):1W.@pl)
mod(commodity(icl>goods):1W.@pl, agricultural(mod<thing):1J)
…
[/S]
[S:8]
…
mod:05(commodity(icl>goods):2N.@pl, agricultural(mod<thing):2A)
and:04(*marketing(icl>commerce)*:1E.@entry, :03)
and:03(consumption(icl>phenomenon):0Y.@entry, production(icl>product):0J.@def)
…
[/S]
[S:13]
…
mod(*marketing(icl>commerce)*:4W, produce(icl>food):6P)
mod(produce(icl>food):6P, :01)
mod(produce(icl>food):6P, agricultural(mod<thing):6C)
and:01(vegetable(icl>food):5S.@entry.@pl, *fruit(pof>plant)*:5H.@pl)
…
[/S]
docid: agro6
relevance: 0.888889
[S:1]
…
and:01(vegetable(icl>food):11.@entry.@pl, *fruit(pof>plant)*:0Q.@pl)
…
[/S]
[S:3]
…

mod(*marketing(icl>commerce)*:04.@def, :01)
and:01(vegetable(icl>food):0W.@entry.@pl,
fruit(pof>plant):0L.@pl)
…
[/S]
[S:8]
…
mod(*marketing(icl>commerce)*:04.@def, :01)
and:01(vegetable(icl>food):0W.@entry.@pl,
fruit(pof>plant):0L.@pl)
…
[/S]
docid: agro4
relevance: 0.166667
[S:1]
…
mod:04(policy(icl>plan):16, *marketing(icl>commerce)*:0W)
mod:04(trade(icl>activity):0I, equity-driven(mod<thing):0A)
…
[/S]
[S:5]
…
pur(demand(icl>thing):04.@def,*vegetable(icl>food>functional thing ,icl>plant>living thing)*:0F.@pl)
…
[/S]

In the above, the bold, italisized text represents the Universal Words in the sentence, which match with the query.

d. Keyword-based Matching

Sample Query: *farm management*
Matched documents:
Total match found: 7
docid: agro1, agro2, agro4, thread334, thread76, thread141, thread183.

e. Meaning-based Search

Sample Query: *performance of agriculture*
UNL of the Query:
mod(performance(icl>operation):00.@entry,agriculture(icl>activity):0F)

Matched UNL documents and relevant sentences therein:
Total match found: 2
docid: agro2
relevance: 1
[S:14]
Though the overall growth of Indian economy has depended much upon the performance of agriculture, over the years, not much public investment has been made on its development.
{unl}
…
obj:02(depend
upon(aoj>thing,obj>thing).@entry.@present.@complete.@although,performance(icl>operation))
man:02(depend
upon(aoj>thing,obj>thing).@entry.@present.@complete.@although,much(icl>how):04)

meaning-based search

mod:02(performance(icl>operation).@def,
agriculture(icl>activity))

obj:03(make(agt>thing,obj>thing).@entry.@present.@complete,investment(icl>assets).@topic)
…
{unl}
[/S]
docid: agro4
relevance: 0.60
[S]
...
obj:05(improve(icl>change(agt>thing,obj>thing)):22.@past,
performance(icl>operation):2B.@entry.@topic)

mod:05(performance(icl>operation):2B.@entry.@topic,
agriculture(icl>activity):2X)

mod:05(agriculture(icl>activity):2X, indian(aoj>thing):2Q)
…
[/S]

Sample Query: *agriculture of performance* [2]
UNL of the Query:

See the difference between UNL expressions of both queries

mod(agriculture(icl>activity):00.@entry,
performance(icl>operation):0F)

Matched UNL documents and relevant sentences therein:
Total match found: 0

In the above, the dotted ellipses represent the expressions in the sentence, which match the query. As can be seen from the results, the system has the capability of distinguishing between two queries having the same keywords but entirely different meanings. For *"performance of agriculture"* the system returns a document but for *"agriculture of performance"* the system cannot find a document, which matches the query. Given these two queries to *keyword-based* search engines, they would return almost the same results.

e. Multilingual Search

Sample Query: खेत-प्रबंधन के उपकरण
UNL of the Query:
mod(tool(icl>functionalthing):13.@entry,
management(icl>activity):0A)
mod(management(icl>activity):0A, farm(icl>place):00)

[2] The reader can see that this is meaningless and should fail the search

Matched UNL documents and relevant sentences therein:
docid: agro4
relevance: 1
[S:22]
खेत-प्रबंधन के लिए संगणक आवश्यक उपकरण बन गए हैं।
{unl}
...
aoj(become(icl>transform{>change}(obj>thing,gol>thing,src>thing)):0F.@entry.@present.@complete.@pred,
computer(icl>machine):00.@pl)
pur(become(icl>transform{>change}(obj>thing,gol>thing,src>thing)):0F.@entry.@present.@complete.@pred,
management(icl>activity):1D)
obj(become(icl>transform{>change}(obj>thing,gol>thing,src>thing)):0F.@entry.@present.@complete.@pred,
tool(icl>functional thing):0Z.@indef)
mod(tool(icl>functionalthing):0Z.@indef,
essential(aoj>thing):0P)
mod(management(icl>activity):1D, farm(icl>place):18)
...
{/unl}
[/S]

The above example illustrates the *multilingual* aspect of the system. Here the input query to the system is given in *Hindi*. The document matched is in *English* and the user has the option of viewing the document in *Hindi*.

The "Fig. 7," shows the *multilingual* aspect of the system.

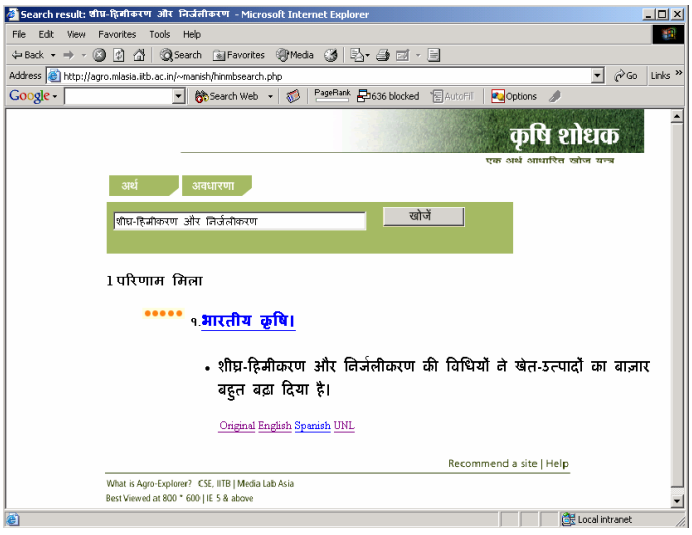

Fig. 7. Search results for Hindi query.

8 Conclusion

We described a *multi stage fall-back search strategy* for cross-lingual information retrieval in the agricultural domain. The results show the effectiveness of the strategy. The system filters out *non-sense* queries and provides high precision in the retrieval. Currently, the quantitative analysis for the performance measurement is not statistically significant, because of the small UNL document base. Therefore the future work consists of

a. Robust and Scalable generation of UNL documents
b. Query Expansion
c. Morphology for Indian languages.

References

[1] Mrugank Surve, Satish Kagathara, Pushpak Bhattacharyya, Agro Explorer Group, *Agro Explorer: a Meaning Based Multilingual Search Engine*, In Proceedings of the International Conference on Digital Libraries (ICDL) 2004, Volume 2, New Delhi, India. http://www.mlasia.iitb.ac.in/docs/agro_icdl.pdf

[2] Sergey Brin and Lawrence Page, *The anatomy of a large-scale hyper textual web search engine,* In Proceedings of the Seventh International World Wide Web Conference (WWW7), 1998. http://decweb.ethz.ch/WWW7/1921/com1921.htm

[3] The *Universal Networking Language (UNL) System* UNL center, UNDL Foundation, 2001. http://www.undl.org/unlsys/index.html

[4] *Enconverter Specifications*, Version 3.1, UNL center, UNDL Foundation, 2001.

[5] *Deconverter Specifications*, Version 3.1, UNL center, UNDL Foundation, 2001.

[6] Paul Buitelaar, Klaus Netter, Feiyu Xu, *Integrating Different Strategies for Cross-Language Information Retrieval in the MIETTA Project,* In Proceedings of TWLT14, Enschede, the Netherlands, December 1998. http://www.mietta.info/docs/mietta-twlt.pdf

[7] W. Bruce Croft, Roger H. Thompson, *The use of adaptive mechanisms for selection of search strategies in document retrieval systems,* In Proceedings of the 7th annual international ACM SIGIR conference on Research and development in information retrieval July 1984. http://portal.acm.org/citation.cfm?id=636811

[8] *Jakarta Lucene* a high-performance, full-featured text search engine. http://jakarta.apache.org/lucene/docs/index.html

[9] http://www.google.com

[10] http://www.oingo.com,http://www.excite.com, http://www.simpli.com

[11] http://www.clush.com