Normalized Distance Measure: A Measure for Evaluating MLIR Merging Mechanisms

Chetana Sidige[1], Sujatha Pothula[1], Raju Korra[1], Madarapu Naresh Kumar[1], Mukesh Kumar[1]

[1] Department of Computer Science, Pondicherry University
Puducherry, 605014, India.

Abstract

The Multilingual Information Retrieval System (MLIR) retrieves relevant information from multiple languages in response to a user query in a single source language. Effectiveness of any information retrieval system and Multilingual Information Retrieval System is measured using traditional metrics like Mean Average Precision (MAP), Average Distance Measure (ADM). Distributed MLIR system requires merging mechanism to obtain result from different languages. The ADM metric cannot differentiation effectiveness of the merging mechanisms. In first phase we propose a new metric Normalized Distance Measure (NDM) for measuring the effectiveness of an MLIR system. We present the characteristic differences between NDM, ADM and NDPM metrics. In the second phase shows how effectiveness of merging techniques can be observed by using Normalized Distance Measure (NDM). In first phase of experiments we show that NDM metric gives credits to MLIR systems that retrieve highly relevant multilingual documents. In the second phase of the experiments it is proved that NDM metric can show the effectiveness of merging techniques that cannot be shown by ADM metric.

Keywords: *Average Distance Measure (ADM), Normalized Distance Measure (NDPM), Merging mechanisms, Multilingual Information Retrieval (MLIR).*

1. Introduction

The Information Retrieval identifies the relevant documents in a document collection to an explicitly stated query. The goal of an IR system is to collect documents that are relevant to a query. Information retrieval uses retrieval models to get the similarity between the query and documents in form of score. Retrieval models are like binary retrieval model, vector space model, and probabilistic model.

Cross-language information retrieval (CLIR) search a set of documents written in one language for a query in another language. The retrieval models are performed between the translated query and each document. There are three main approaches to translation in CLIR: Machine translation, bilingual machine-readable dictionary, Parallel or comparable corpora-based methods.

Irrelevant documents are retrieved by information retrieval model when translations are performed with unnecessary terms. Thus translation disambiguation is desirable, so that relevant terms are selected from a set of translations. Sophisticated methods are explored in CLIR for maintain translation disambiguation part-of-speech (POS) tags, parallel corpus, co-occurrence statistics in the target corpus, the query expansion techniques. Problem called language barrier issues raised in CLIR systems [2].

Due to the internet explosion and the existence of several multicultural communities, users are facing multilingualism. User searches in multilingual document collection for a query expressed in a single language kind of systems are termed as MLIR system. First, the incoming question is translated into target languages and second, integrates information obtained from different languages into one single ranked list. Obtaining rank list in MLIR is more complicated than simple bilingual CLIR. The weight assigned to each document (RSV) is calculated not only according to the relevance of the document and the IR model used, but also the rest of monolingual corpus to which the document belongs is a determining factor.

Two types of multilingual information retrieval methods are query translation and document translation. As document translation causes more complications than query translation, our proposal is applying query translation. Centralized MLIR and distributed MLIR are two type architectures. Our proposed metric is applied on distributed MLIR. Distributed MLIR architecture has problems called merging the result lists. Merging techniques are like raw score, round robin. Performance of MLIR system differs due to merging methods. To measure the MLIR performance correctly we need to consider the MLIR features like translation (language barrier), merging methods. Our new metric is based on the concept of ADM metric. The drawbacks of the ADM metric are overcome in the proposed formula.

In this paper, Section 2 explains the related work of the proposed metric and merging methods. Section 3 explains the proposed metric in two phases. First phase explains newly proposed metric and second phase explains how the proposed metric is applied for merging methods of MLIR. Section 4 explains the experimental results and section 5 states conclusion.

2. Related work

There are two types of translation methods in MLIR - query translation and document translation [2]. Document translation can retrieve more accurate documents than query translation because the translation of long documents may be more accurate in preserving the semantic meaning than the translation of short queries. Query translation is a general and easy search strategy.

There are two architectures in MLIR [12]. In centralized architecture consists of a single document collection containing document collections and a huge index file. It needs one retrieving phase. Advantage of centralized architecture is it avoids merging problem. Problem with centralized architecture is the weights of index terms are over weighting. Thus, centralized architecture prefers small document collection. In distributed architecture, different language documents are indexed in different indexes and retrieved separately. Several ranked document lists are generated by each retrieving phase. Obtaining a ranked list that contains documents in different languages from several text collections is critical; this problem is solved by merging strategies. In any architecture problem called language translation issues are raised.

In a distributed architecture, it is necessary to obtain a single ranked document list by merging the individual ranked lists that are in different languages. This issue is known as merging strategy problem or collection fusion problem. Merging problem in MLIR is more complicated than the merging problem in monolingual environments because of the language barrier in different languages.

Following are some of the merging strategies.

Round-robin merging strategy: This approach is based on the idea that document scores are not comparable across the collections, each collection has approximately the same number of relevant documents and the distribution of relevant documents is similar across the result lists [11]. The documents are interleaved according to ranking obtained for each document.

Raw score merging strategy: This approach is based on the assumption that scores across different collections are comparable. Raw score sorts all results by their original similarity scores and then selects the top ranked documents. This method tends to work well when same methods are used to search documents [11].

Normalized score merging: This aprroch is based on the assumption that merging result lists are produced by diverse search engines. A simplest normalizing approach is to divide each score by the maximum score of the topic on the current list. After adjusting scores, all results are sorted by the normalized score [10], [11]. Another method is to divide difference between the score and maximum score by difference between maximum score and minimum score. This type of merging favours the scores which are near the best score of the topic on the list. This approach maps the scores of different result lists into the same range, from 0 to 1, and makes the scores more comparable. But it has a problem. If the maximum score is much higher than the second one in a result list, the normalized-score of document at rank 2 would be low even if its original score is high.

System evaluation is measured by calculating gap between system and user relevance. Due to Lack of control variables measuring the user centered approach is becoming difficult. The motivation of our proposal is performance measurement can be examined by the agreement or disagreement between the user and the system rankings.

New metric NDM is generated by considering the features of below IR metrics.

Discount Cumulated Gain (DCG): As rank gets increased the importance of document gets decreased.

Normalized Distance-based Performance Measure (NDPM): NDPM gives performance of MLIR system by comparing the order of ranking of two documents [1] [5]. NDPM is based on a preference relation $\succ$ on a finite set of documents D is a weak order.

Average Distance Measure (ADM): [3] ADM measures the average distance between UREs (user relevance estimation) (the actual relevances of documents) and SREs (system relevance estimation) (their estimates by the IRS) [2]. Drawback of ADM metric is low ranked documents are given equal importance high ranked documents [3][1]. Problem with precision and recall is, they are highly sensitive to the thresholds. Instead of changing the relevance, retrieval values suddenly, there should be a continuous varying of relevance and retrieval.

3. Proposed metric

Normalized Distance Measure (NDM) is a new metric designed mainly for evaluating MLIR system. MLIR system has to access more information in an easier and faster way than monolingual systems. Distributed MLIR system has three steps translation, retrieval and merging. NDM considers ranking as a suitable measurement, because continuous rank performance measurement is better than non continuous groping and also the document score of one language cannot be compared to another language. Normalized Distance Measure measures the difference between the user's estimated ranked list and final MLIR ranked list. The NDM value ranges from 0 to 1. Final rank list of MLIR represented as R_{MLIR}. The ranked list obtained from user is represented as R_{USER}.

$$NDM = 1 - \frac{\sum_{i=0}^{m} \left| \frac{R_{MLIR(i)} - R_{USER(i)}}{R_{MLIR(i)} + \alpha} \right|}{\sum_{i=0}^{m} \left| \frac{R_{Threshold(i)} - R_{USER(i)}}{R_{Threshold(i)} + \alpha} \right|} \quad (1)$$

Where i = {0, 1, 2… m} where m is total number of documents.

In (1) equation, the term $R_{MLIR(i)} + \alpha$ is total penalty calculated. 'α' is included in (1) equation the penalty when an relevant document is not retrieved or when non relevant document is retrieved. Penalty R_{MLIR} measures the precision

Six cases are as follows.

Case (a): $R_{MLIR(i)} = R_{USER(i)}$

Case (b): $R_{MLIR(i)} > R_{USER(i)}$

Case (c): $R_{MLIR(i)} < R_{USER(i)}$

Case (d): $R_{MLIR(i)} = 0, R_{USER(i)} = 0$

Case (e): $R_{MLIR(i)} \neq 0, R_{USER(i)} = 0$

Case (f): $R_{MLIR(i)} = 0, R_{USER(i)} \neq 0$

First three cases consider a document as relevant by both MLIR system and USER. Last three cases a document is considered as not relevant by either MLIR system or by USER. In case (a), (d) difference between rankings is 0 as both ranks are same. In case (c), (f) difference between rankings is positive. This is represented on left bottom of the diagonal in table 1. In case (b), (e) difference between rankings is negative. This is represented on top right of the diagonal in table 1.

Table 1. Calculation of Distance Between MLIR and USER Rank Systems In All Six Possibilities

	$R_{MLIR(i)}$	**0**	**1**	**2**	**3**	**4**	**5**
$R_{USER(i)}$	$R_{MLIR(i)} + \alpha$ / $R_{USER(i)} + \alpha$	**1**	**2**	**3**	**4**	**5**	**6**
0	**1**	0	0.5	0.67	0.75	0.8	0.83
1	**2**	1	0	0.33	0.5	0.6	0.67
2	**3**	2	0.5	0	0.25	0.4	0.5
3	**4**	3	1	0.33	0	0.25	0.33
4	**5**	4	1.5	0.67	0.25	0	0.17
5	**6**	5	2	1	0.5	0.25	0

We can estimate the good MLIR System by using the user estimated values but estimating a worst MLIR is not possible because worseness of MLIR system increases as the irrelevant documents are increased. Thus we are using threshold MLIR as a least bad case MLIR system. The denominator measures the difference between the resulted ranked lists and threshold MLIR system. The numerator measures the difference between the MLIR ranked list and ranked list estimated by user.

Table 2 shows the different characteristics of ADM, NDPM and NDM. In Table 2, the characteristic called "document score" is not needed for user. User is concerned only about ordering and ranking of the document list. NDM gives different importance for first and last documents. other characteristics shows the reasons, why NDM metric is performing better than other metrics.

Table 2: Characteristics of NDM, ADM, NDPM

Characteristics	ADM	NDPM	NDM
Rank	No	No	Yes
Order	No	Yes	Yes
Document score	Yes	No	No
Considers irrelevant document	Yes	No	Yes
Equal Importance for first and last documents	Yes	Yes	No

4. Experimental results

Phase 1 experiments show the importance of NDM metric. Effectiveness of an Information Retrieval System (IRS) depends on relevance and retrieval. [2] States that precision and recall are highly sensitive to the thresholds chosen.

Table 3: Document scores in six MLIR systems

	D1	***D2***	***D3***	***D4***	***D5***
USER	0.9	0.8	0.7	0.6	0.5
MLIR1	**0.8**	**0.7**	**0.6**	**0.5**	0.9
MLIR2	0.9	**0.7**	**0.6**	**0.8**	0.5
MLIR3	0.9	**0.6**	**0.8**	**0.7**	0.5
MLIR4	0.9	0.8	0.7	**0.5**	**0.6**
MLIR5	**0.8**	**0.9**	0.7	0.6	0.5
MLIR6	0.9	**0.7**	**0.8**	**0.5**	**0.6**

Precision and recall are not continuous therefore precision and recall are not sensitive to important changes to MLIR systems like giving importance to top relevant documents. ADM and NDPM metrics are continuous metrics. Thus we are comparing the NDM metric with ADM and NDPM.

Table 4: Compare NDM with ADM and NDPM

	ADM	***NDPM***	***NDM***
MLIR1	0.84	0.60	0.647
MLIR2	0.92	0.80	0.863
MLIR3	0.92	0.80	0.885
MLIR4	0.96	0.90	0.9507
MLIR5	0.96	0.90	0.9554
MLIR6	0.92	0.80	0.987

Table 3 represents the six MLIR system's score list. The scores of the document are converted into rankings to obtain NDM and NDPM metrics. The drawbacks of the ADM are stated in [3]. The drawbacks of ADM are corrected in NDM. [3] states the importance of ranking in performance measurement. Table 4 compares NDM metric with ADM and NDPM.

We ordered 6 MLIR systems in Table 3 in such a way that the bottom MLIR system performance is better than the top MLIR systems. In Table 4 the ADM and NDPM values of the 6th MLIR system is low even though its performance is better that 4th and 5th MLIR system. Distribution of relevant documents is slightly different in MLIR3 and MLIR4, so NDM values are slightly different but ADM and NDPM shows no difference in performance. In MLIR2 and MLIR3 2nd, 3rd, 4th documents are interchanged among themselves. MLIR1 gives bad performance because the 1st top document is placed at last position. Figure 1 represents the table 4.

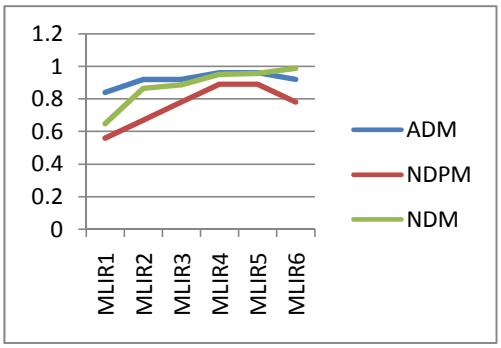

Fig. 1 The performance of NDM is compared with the ADM and NDPM

In the second phase of our experiments, we have measured the NDM values for four merging technique of a MLIR system. ADM value for the above MLIR system is 0.68 which remains constant for all 4 merging techniques. To obtain the performance of merging mechanisms of an MLIR we use NDM metric as follows. We took 9 documents from 3 languages and assigned document scores for 9 documents as shown in Table 5.

Table 5: Scores of 9 documents in three languages

Language 1	Language 2	Language 3
1.9	0.4	1.2
1.62	0.2	0.9
1.4		0.6
0.8		

We performed merging techniques for the above MLIR and the documents order is shown in the table 6. The ADM and NDM values for four merging mechanisms are shown in the Table 7.

Table 6: Rank lists of merging techniques

rank	Round robin	Raw score	Normalize with max(RSV)	Normalize with max(RSV) and min(RSV)
1	1.9	1.9	1	2
2	0.4	1.62	1	2
3	1.2	1.4	1	1.72
4	1.62	1.2	0.8	1.5
5	0.2	0.9	0.75	1.4
6	0.9	0.8	0.73	1.2
7	1.4	0.6	0.5	1
8	0.6	0.4	0.5	1
9	0.8	0.2	0.421	1.72

Table 7: NDM measure for 9 documents in three languages.

	ADM	NDM
Round Robin Merging	0.68	0.88
Raw Score Merging	0.68	0.84
Normalized score merging with max (RSV)	0.68	0.95
Normalized score merging with max (RSV) and min (RSV)	0.68	0.85

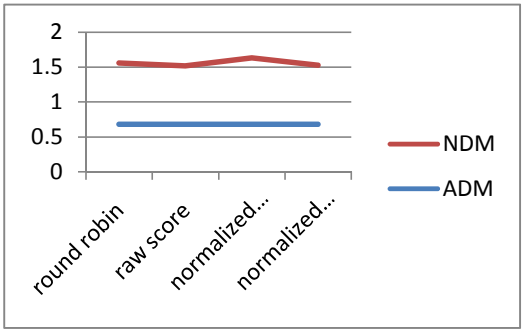

Fig 2: Graphical representation of table r.

Fig 2 shows the variation of NDM metric for merging techniques, where ADM shows no difference. Characteristics of the NDM, ADM, NDPM shows that NDM considered many features.

5. Conclusions

This paper shows two phased experiment where first phase proposes a new metric for MLIR based on rank schema. It is shown that the new meteric is better than old metrics like ADM and NDPM metrics. Characteristics that differentiate three metrics ADM, NDPM and NDM are tabularized. In the first phase we stated the benefits of NDM over ADM and NDPM in form of characteristics and experiments. In the second phase NDM metric evaluates the performance of MLIR system when four different types of merging techniques are used.

References

[1] Bing Zhou, Yiyu Yao, "Evaluating Information Retrieval System Performance Based on User Preference", Journal of intelligent information systems, Springerlink, vol. 34, issue 3, pp. 227-248, June. 2010.

[2] Kazuaki Kishida, "Technical issues of cross-language information retrieval: a review", Information Processing and Management international journal, science direct, vol. 41, issue 3, pp. 433-455, may. 2005.

[3] Stefano Mizzaro, S, "A New Measure of Retrieval Effectiveness (Or: What's Wrong with Precision and Recalls)," In: International Workshop on Information Retrieval, pp. 43–52 .2001

[4] Järvelin, K., Kekäläinen, J.: Cumulated Gain-based Evaluation of IR Techniques. ACM Transactions on Information Systems, vol. 20, Issue 4, 422—446 October (2002)

[5] Yao, Y. Y. (1995). Measuring retrieval effectiveness based on user preference of documents. Journal of the American Society for Information Science, Volume 46 Issue 2, 133–145, March 1995.

[6] W. C. LIN and H. H. CHEN, "Merging results by using predicted retrieval effectiveness," Lecture notes in computer science, pages 202–209, 2004.

[7] Savoy, "Combining multiple strategies for effective monolingual and cross-lingual retrieval," IR Journal, 7(1-2):121-148, 2004.

[8] Lin, W.C. & Chen, H.H. (2002b). Merging Mechanisms in Multilingual Information Retrieval. In Peters, C. (Ed.), Working Notes for the CLEF 2002 Workshop, (pp. 97-102).

[9] Rita M. Aceves-Pérez, Manuel Montes-y-Gómez, Luis Villaseñor-Pineda, Alfonso Ureña-López. Two Approaches for Multilingual Question Answering: Merging Passages vs. Merging Answers International Journal of Computational Linguistics and Chinese Language Processing. Vol. 13, No. 1, pp 27-40, March 2008.

[10] F. Mart´ınez-Santiago, M. Mart´ın, and L.A. Ure˜na. SINAI at CLEF 2002: Experiments with merging strategies. In Carol Peters, editor, Proceedings of the CLEF 2002 Cross-Language Text Retrieval System Evaluation Campaign. Lecture Notes in Computer Science, pages 103–110, 2002.

[11] E. Airio, H. Keskustalo, T. Hedlund and A. Pirkola, Multilingual Experiments of UTA at CLEF2003 - the Impact of Different Merging Strategies and Word Normalizing Tools. CLEF 2003, Trondheim, Norway, 21-22 August 2003.

[12] Wen-Cheng Lin and Hsin-Hsi Chen (2003). Merging Mechanisms in Multilingual Information Retrieval. In Advances in Cross-Language Information Retrieval: Third Workshop of the Cross-Language Evaluation Forum, CLEF 2002, Lecture Notes in Computer Science, LNCS 2785, September 19-20, 2002, Rome, Italy, pp. 175-186.

[13] Anne Le Calvé, Jacques Savoy, "Database merging strategy based on logistic regression," Information Processing and Management: an International Journal, vol.36, p.341-359, May. 2000.

Chetana Sidige is presently pursuing M.Tech (Final year) in Computer Science of Engineering at Pondicherry University. She did her B.Tech in Computer Science and Information Technology from G. Pulla Reddy Engineering College, Sri Krishnadevaraya University. Currently the author is working on Multilingual Information retrieval evaluation.

Pothula Sujatha is currently working as Assistant Professor and pursuing her PhD in Department of Computer science from Pondicherry University, India. She completed her Master of Technology in Computer Science and Engineering from Pondicherry University and completed her Bachelor of Technology in Computer Science and Engineering from Pondicherry Engineering College, Pondicherry. Her research interest includes Modern Operating Systems, Multimedia Databases, Software Metrics and Information Retrieval. Her PhD research is on performance Evaluation of MLIR systems.

Raju Korra is presently pursuing Master of Technology in Computer Science and Engineering from Pondicherry University, India. He has completed his Bachelor of Technology in Computer Science and Engineering from Kakatiya University, Warangal. His research interest includes Genetic Algorithms, Software metrics, Data Mining, Information Retrieval and MLIR. Currently he is working on metrics for evaluating MLIR systems.

Madarapu Naresh Kumar is presently pursuing Master of Technology in Computer Science with specialization in Network and Internet Engineering from Pondicherry University, India. He has completed his Bachelor of Technology in Computer Science and Engineering from JNTU Hyderabad. His research interest includes Cloud Computing, Web Services, Software Metrics, SOA and Information Retrieval. Currently he is working on security issues in Cloud Computing.

Mukesh Kumar received his Bachelor of Technology degree in Computer Science and Engineering from Uttar Pradesh Technical University Lucknow, India in 2009. He is currently pursuing his master's degree in Network and Internet Engineering in the School of Engineering and Technology, Department of Computer Science, Pondicherry University, India. His research interests include Denial-of Service resilient protocol design, Cloud Computing and Peer to Peer Networks.