Combining Resources with Confidence Measures for Cross Language Information Retrieval

Youssef Kadri
Laboratoire RALI, DIRO
Université de Montréal
CP 6128, Succ Centre ville, Montréal, Canada, H3C3J7
1-514-343 6111 Ext. 1651
kadriyou@iro.umontreal.ca

Jian-Yun Nie
Laboratoire RALI, DIRO
Université de Montréal
CP 6128, Succ Centre ville, Montréal, Canada, H3C3J7
1-514-343 2263
nie@iro.umontreal.ca

ABSTRACT
Query translation in Cross Language Information Retrieval (CLIR) can be performed using multiple resources. Previous attempts to combine different translation resources use simple methods such as linear combination. Unfortunately, these approaches are insufficient to combine different types of resources such as bilingual dictionaries and statistical translation models. In this paper, we use confidence measures for this combination for the purpose of English-Arabic CLIR. Confidence measure is used to adjust the original scores of translations and to create a weight of the same nature for translations with different resources. We tested this technique on two test CLIR collections from TREC and obtained encouraging improvements compared to the results of linear combination.

Categories and Subject Descriptors
H.3.3 [Information storage and retrieval]: Information Search and Retrieval – *retrieval models, query formulation.*

General Terms
Algorithms, Performance, Experimentation, Theory.

Keywords
CLIR, confidence measures, linear combination.

1. INTRODUCTION
The goal of Cross Language Information Retrieval (CLIR) is to retrieve relevant documents written in a language different from the query language. In CLIR, the key problem is query translation. To translate queries, one can use Bilingual Dictionaries (BDs), parallel corpora or machine translation [10]. Good performance has been obtained when combining multiple translation resources especially parallel corpora and BDs [9]. As different resources may suggest different translations, it is better to combine them so as to obtain as many correct translations as possible. The fact that more correct translations are provided for a query leads naturally to a query expansion effect, which is desirable in Information Retrieval (IR). Appropriate term weighting is another crucial problem in CLIR. If Statistical Translation Models (STMs) are used for query translation, the probability of translation can be used as the weight of the term. On the other hand, as BDs do not provide probabilities to translations, weights are usually determined according to a uniform distribution or according to their occurrences and co-occurrences in a corpus [11]. When several translation tools or resources are combined, a crucial problem is to combine all the translation candidates correctly. In the previous studies, simple methods are usually employed, i.e. one combines various translations for the same query term linearly by assigning [12] or optimizing automatically [9] a confidence weight to the translation tool or resource. However, we notice that a single weight is assigned to each translation resource. It does not modify the relative importance of the translations from the same resource. In practice, sometimes when new criteria are considered, a translation with a low score suggested by a resource can turn out to be a better translation. In this case, it is important to modify the relative importance of this translation in the same set of translations. For example in a TREC 2001 query, the word "develop" in "to develop tourism in Cairo" (لتطوير السياحه في القاهره) is translated into Arabic by a STM with the following set:

{تنميه 0.48 (development), نامي 0.13 (developed), إنماء 0.08 (development), تطور 0.06 (evolution), تطوير 0.04 (development)}.

We observe that the most common translation word "تطوير" (development) only takes the fifth place with much lower probability than "تنميه". If a linear combination is used to combine this translation model with another resource (say a BD), it is unlikely that the correct translation word "تطوير" could gain larger weight than "تنميه". It is then important to reconsider each translation candidate according to additional criteria in order to produce a new score for it. In so doing, the initial ranking of translation candidates can be changed. As a matter of fact using the method of confidence measures we propose in this paper, we are able to reorder the translation candidates as follows:

{تطوير 0.51, تنميه 0.29}.

The weight of the correct translation "تطوير" is considerably increased. Confidence measure technique can adjust the weight of possible translations of a query term according to additional informative features. This confidence measures our certainty that the translation is correct. It provides a means to re-weight the translation candidates in a homogenous manner for translation candidates from different resources. Therefore, the advantages of this approach are twofold. On one hand, the confidence measure allows us to adjust the original weight of the translations and to

PIKM'07, November 9, 2007, Lisbon, Portugal.

select the best translation terms. On the other hand, the confidence estimates also provide us with a comparable weighting for the translation candidates across different translation resources. Consequently, confidence measure can be viewed as a general mechanism to combine effectively different translation resources. Our experiments also show that this method outperforms the linear combination method on the two test collections.

The remainder of the paper is organized as follows: Related works to confidence measures are described in section 2. Section 3 presents the mechanism of integrating these confidence measures in CLIR. It will be followed by the description of the process of computing confidence measures. Finally, we present the results of our experiments with analysis.

2. BACKGROUND

Confidence estimation was originally used in speech recognition and understanding [6]. Because errors occur frequently in speech recognition, an accurate confidence measure can help to determine if the recognition result is correct. Confidence estimation has been applied to improve recognition by incorporating extra information into the recognition process. Introducing confidence scores contributed substantially to the reduction of the recognition error rate. This performance is due to the fact that when low confidence is attributed to a word hypothesis, the latter is often a wrong recognized word according to the extra information. Therefore, it can be rejected. Gandrabur and al. [5] used confidence measures in a translation prediction task. They used neural nets to estimate the conditional probability of correctness $p(C=1 \mid w_m, h, s)$ for a prediction w_m which follows the history h in the translation of a source sentence s. Here as well, a significant gain is observed when using a confidence estimation layer within the translation models [5].

In CLIR, we observe the same problem as in speech recognition: query translation can be performed with many resources for the same word. Furthermore, the translations suggested by different resources are assigned different and often incompatible weights. Therefore, we are provided with different translation alternatives with different probabilities. It is necessary to combine these alternatives in order to select the best ones. As in speech recognition and machine translation, confidence measures can be used to learn how to adjust the original scores of translations by observing their performance on new texts. These confidence estimates will be used in this paper as a uniform measure on translations instead of the original probabilities. Concretely, for a given translation produced by any resource, STM or BD, we aim to measure the confidence of it being correct, according to some informative features (Section 4.3).

3. INTEGRATING CONFIDENCE MEASURES IN CLIR

Let us describe the general framework of CLIR that integrates confidence measures. We use a retrieval model based on language modeling. Given a query Q_E written in a source language E and a document D_A represented in a target language A, we can compute the relevance of this document to the query with the negative of the divergence of the query's language model from the document's language model [13]:

$$R(Q_E, D_A) \propto \sum_{t_A} p(t_A \mid Q_E) \log p(t_A \mid D_A) \qquad (1)$$

To avoid the problem of attributing zero probability to query terms not occurring in document D_A, smoothing techniques are used to estimate $p(t_A|D_A)$. One can use the Jelinek-Mercer smoothing technique which is a method of interpolating between the document and collection language models [14]. The smoothed $p(t_A|D_A)$ is calculated as follows:

$$p(t_A \mid D_A) = (1-\lambda) p_{ML}(t_A \mid D_A) + \lambda p_{ML}(t_A \mid C_A) \qquad (2)$$

where $p_{ML}(t_A \mid D_A) = \frac{tf(t_A, D_A)}{|D_A|}$ and $p_{ML}(t_A \mid C_A) = \frac{tf(t_A, C_A)}{|C_A|}$ are the maximum likelihood estimates of a unigram language model based respectively on the given document D_A and the collection of documents C_A. λ is a parameter that controls the influence of each model.

The term $p(t_A \mid Q_E)$ in equation (1) representing the query model can be estimated in the source language by:

$$\begin{aligned} p(t_A \mid Q_E) &= \sum_{q_E} p(t_A, q_E \mid Q_E) \\ &= \sum_{q_E} p(t_A \mid q_E, Q_E) p(q_E \mid Q_E) \\ &\approx \sum_{q_E} p(t_A \mid q_E) p_{ML}(q_E \mid Q_E) \end{aligned} \qquad (3)$$

where $p_{ML}(q_E \mid Q_E)$ is the maximum likelihood estimation: $p_{ML}(q_E \mid Q_E) = \frac{tf(q_E, Q_E)}{|Q_E|}$ and $p(t_A \mid q_E)$ is the translation model. Replacing (3) in (1), we obtain the general ranking formula:

$$R(Q_E, D_A) \propto \sum_{t_A} \sum_{q_E} p(t_A \mid q_E) p_{ML}(q_E \mid Q_E) \log p(t_A \mid D_A) \qquad (4)$$

Our work focuses on the estimation of the translation model $p(t_A \mid q_E)$. Traditionally when translation is done with more than one resource, linear combination is used to estimate the translation model as follows:

$$p(t_A \mid q_E) = z_{q_E} \sum_i \lambda_i p_i(t_A \mid q_E) \qquad (5)$$

where λ_i is the parameter related to the translation resource i and z_{q_E} is a normalization factor so that $\sum_{t_A} p(t_A \mid q_E) = 1$. The parameters λ denote the confidence weight assigned to each resource. These parameters can be optimized using some training data. $p_i(t_A \mid q_E)$ is the probability of translating the source

word q_E to the target word t_A by the resource *i*. As we discussed earlier, this method assigns a weight to each resource. So the question dealt with is: Given a translation *resource*, how much trust should we place on it? Instead, the question we have to ask is: Given a translation *candidate*, is it correct and how confident are we on its correctness?

Confidence measure is used to answer this very question. We use confidence measure to reconsider each of the translation candidates according to additional features. Given a translation candidate t_A for a source term q_E, $p(t_A \mid q_E)$ is computed with the sum of confidence estimates on this candidate using different resources, i.e.

$$p(t_A \mid q_E) = z_{q_E} \sum_i p_i(C=1 \mid t_A, q_E, F) \quad (6)$$

where F is the set of features that we use, $p(C=1 \mid t_A, q_E, F)$ is the probability of correctness of t_A for translating q_E. This probability is normalized such that: $\sum_{t_A} p(C=1 \mid t_A, q_E, F) = 1$.

4. COMPUTATION OF CONFIDENCE MEASURES

Confidence for a translation is defined as the posterior probability that this translation is correct *P(C=1|X)*, given *X*— the source word, a translation and a set of features. As the output is either *C=1* (correct) or *C=0* (incorrect), we can use a binary classifier to determine *P(C=1|X)*.

4.1 Learning confidence with MLP

We use a Multi Layer Perceptron (MLP) to estimate the probability of correctness *P(C=1|X)* of a translation. Neural networks have the ability to use input data of different natures and they are well-suited for classification tasks.

Our training data can be viewed as a set of pairs *(X,C)*. Where *X* is a vector of features relative to a translation[1] used as the input of the network, and *C* is the desired output (the correctness of the translation 0/1). The MLP implements a non-linear mapping of the input features by combining layers of linear transformation and non-linear transfer function. Formally, the MLP implements a discriminant function for an input X of the form:

$$g(X;\theta) = o(V \times h(W \times X)) \quad (7)$$

where θ ={W,V}, W is a matrix of weights between input and hidden layers and V is a vector of weights between hidden and output layers; *h* is an activation function for the hidden units which non-linearly transforms the linear combination of inputs $W \times X$; *o* is also a non-linear activation function but for the output unit, that transforms the MLP output to the probability estimate *P(C=1|X)*. Under these conditions, our MLP was trained to minimize an objective function of error rate (Section 4.2). In our experiments, we used a batch gradient descent optimizer.

[1] By translation, we mean the pair: source word and its translation.

During the test stage, the confidence of a translation X is estimated with the above discriminant function *g(X; θ)*; where θ is the set of weights optimized during the learning stage. These parameters are expected to correlate with the true probability of correctness *P(C=1|X)*.

4.2 The objective function to minimize

The training and test data are pairs of sentences that are considered to be mutual translations. The objective function aims to reflect the correspondence between these sentences. A natural metric for evaluating probability estimates is the negative log-likelihood (or cross entropy CE) assigned to the test corpus by the model normalized by the number of examples in the test corpus [2]. This metric evaluates the probabilities of correctness. It measures the cross entropy between the empirical distribution on the two classes (correct/incorrect) and the confidence model distribution across all the examples $X^{(i)}$ in the corpus. Cross entropy is defined as follows:

$$CE = -\tfrac{1}{n} \sum_i \log P(C^{(i)} \mid X^{(i)}) \quad (8)$$

where $C^{(i)}$ is 1 if the translation $X^{(i)}$ is correct, 0 otherwise. To remove dependence on the prior probability of correctness, Normalized Cross Entropy (NCE) is used:

$$NCE = (CE_b - CE)/CE_b \quad (9)$$

The baseline CE_b is a model that assigns fixed probabilities of correctness based on the empirical class frequencies:

$$CE_b = -(n_0/n)\log(n_0/n) - (n_1/n)\log(n_1/n) \quad (10)$$

where n_0 and n_1 are the numbers of correct and incorrect translations among *n* cases in the test corpus.

4.3 Features

The MLP tends to capture the relationship between the correctness of the translation and the features, and its performance depends on the selection of informative features. These features are used together for estimating confidence. In our work, we selected intuitively seven classes of features hypothesized to be informative for the correctness of a translation.

Translation model index: an index representing the resource of translation. In our case, we use four models: a STM built on a set of parallel Web pages [7], another STM built on the English-Arabic United Nations corpus [4], Ajeeb[2] bilingual dictionary and Almisbar[3] bilingual dictionary.

The two STMs are trained using GIZA [1]. The UN corpus is built manually from the United Nations archives. It contains 38 000 pairs of documents. The Web pages corpus is collected from the Web automatically [7]. Its size is 2 816 pairs of documents. The other resources used for translation are English-Arabic bilingual dictionaries: Ajeeb BD includes 20K entries and Almisbar BD 11K entries.

[2] http://www.ajeeb.com/
[3] http://www.almisbar.com/

Translation probabilities: the probability of translating a source word with a target word. These probabilities are estimated with IBM model 1 [3] on parallel corpora. For BDs, as no probability is provided, we carry out the following process to assign a probability to each translation pair *(e,a)* in the BD: We trained a STM on a parallel corpora provided by LDC[4]. Then for each translation pair *(e,a)* of the BD, we looked up the resulting STM and extracted the probability assigned by this STM to the translation pair in question. Finally, the probability is normalized by the Laplace smoothing method:

$$p_{BD}(a \mid e) = \frac{p_{STM}(a \mid e) + 1}{\sum_{i=1}^{n} p_{STM}(a_i \mid e) + 1} \quad (11)$$

Where *n* is the number of translations proposed by the BD to the word *e*.

Translation ranking: This class includes two features: The rank of the translation provided by each resource and the probability difference between the translation and the highest probability translation.

Reverse translation information: This includes the probability of translation of a target word to a source word. Other features measure the rank of source word in the list of translations of the target word and if the source word holds in the best translations of the target word.

Translation "Voting": This feature aims to know whether the translation is voted by more than one resource. The more a same translation is voted the more likely it may be correct.

Source sentence-related features: Features in this class aim to capture the translation relation between the source sentence words and the translation in question. One feature measures the frequency of the source word in the source sentence. Another feature measures the number of source words in the source sentence that have a translation relation with the translation in question.

Language model features: We use the unigram, the bigram and the trigram language models for source and target words on the training data.

4.4 Experiments on confidence measures

4.4.1 Confidence training data

The implementation of the confidence model requires a collection of training data. This data must be different from the one used to train our baseline models. Our training corpus is the Arabic-English parallel news acquired from LDC. It consists of around 83 K pairs of aligned sentences. Source (English) sentences are translated to Arabic word by word using baseline models (2 STMs and 2 BDs). We translated each source word with the most probable[5] translations for the STMs and the best five translations provided by the BDs. Translations are then compared to the reference sentence to build a labeled corpus. However, we do not have the exact translation relationships between words in the corresponding sentences. Therefore, a simple approach is used for the creation of a labeled corpus: a translation of a source word is considered to be correct if it occurs in the reference sentence. The word order is ignored, but the number of occurrences is taken into account. This metric fits well our context of IR: IR models are based on "bag of words" principle and the order of words is not considered.

4.4.2 Impact of hidden units

The number of hidden nodes in MLP usually has an impact on the performance. We test with various numbers of hidden units. The table below shows the performance of these various architectures as measured by the evaluation metric (NCE) on our test dataset. The NCE measures the relative drop in negative log-likelihood compared to the baseline that depends on the prior probability of correctness. The higher NCE, the better the performance. All these experiments are conducted with the free machine learning library Plearn[6].

Table1. Results of several MLP architectures

# hidden units	NCE
5	62.58
10	62.56
20	62.58
50	62.64
100	62.41

Table 1 shows the improvement in cross entropy compared to the baseline model described in section 4.2. According to these results, the difference in performance seems rather minor. Even if the MLP with 50 hidden units gave the best performance, it is difficult to determine a clear pattern of performance with number of hidden units.

4.4.3 Impact of individual features

To test the performance of individual features on the test set, we experimented with each class of features alone. The following table shows the results.

Table2. Feature performance on the test set

Features	NCE
Source sentence-related features	32.20
Translation model index	33.03
Reverse translation information	36.93
Translation ranking	38.28
Translation probabilities	44.46
LM features	50.89
Translation "voting"	57.63
All features	62.56

[4] http://www.ldc.upenn.edu/
Arabic-English Parallel News Part 1 (LDC2004T18)

[5] The translations with the probability p(a|e)≥0.1

[6] http://plearn.berlios.de/

From this table, we can see that the best features are the translation "voting", language model features and the translation probabilities. The translation "voting" is very informative because it presents the translation probability attributed by each resource to the translation in question. The translation ranking, the reverse translation information, the translation model index and the source sentence-related features provide some marginally useful information.

5. ENGLISH-ARABIC CLIR EXPERIMENTS

In order to validate our confidence model for CLIR, we use English queries to retrieve Arabic documents. In our experiments, we used the Arabic TREC collection[7] which contains 383 872 documents, selected from AFP (France Press Agency) Arabic Newswire. These documents are newspaper articles covering the period from May 1994 to December 2000. We use two sets of topics: TREC2001 (25 queries) and TREC2002 (50 queries). All topics have three parts: title, description and narrative. We used only the title and description parts of the topics in our experiments.

During indexing, documents and queries are stemmed and stop-words are removed. The Porter technique is used to stem English queries. Arabic documents are stemmed using linguistic-based stemming method [8]. The query terms are translated with the two STMs and the two BDs. The resulting translations are then submitted to the information retrieval process. We tested with different ways to assign weights to translation candidates: Original translation probabilities of each resource, linear combination and confidence measures. In the next sections, we present the results of experimentation with some analysis.

5.1 Classical models

For query translation, we operated with the four resources separately and then combined them:

Individual models: When using each resource separately, we attribute the IBM 1 translation probabilities to our translations as weights. For each query term, we take only translations with the probability $p(a|e) \geq 0.1$ when using STMs and the five best translations when using BDs.

Linear combination (LC): We use a linear combination to combine the four resources. Each model is assigned a coefficient denoting our confidence in it. The coefficients are optimized on a validation set C of parallel sentences (English-Arabic aligned sentences), by using the EM algorithm to find values which maximize the likelihood LL of this set of data according to the combined model:

$$LL(C) = \sum_{(a,e)} p(a,e) \sum_{j=1}^{|a|} \log \sum_{k=1}^{4} \sum_{i=1}^{|e|} \lambda_k t_k(a_j \mid e_i) p(e_i) \quad (12)$$

Where $p(a,e) = \frac{\#(a,e)}{|C|}$ is the prior probability of the pair of sentences (a,e) in the corpus C, $|a|$ is the length of the target sentence a and $|e|$ is the length of the source sentence e. λ_k is the coefficient related to resource k that we want to optimize. $t_k(a_j|e_i)$ is the probability of translating the source word e_i with the target word a_j with each resource. $p(e_i)$ is the prior probability of the source word e_i in the corpus C. The tuned parameters assigned to each model are as follows:

- STM trained on the Web pages: 0.2927,
- STM built on the UN corpus: 0.3369,
- Ajeeb BD: 0.1457,
- Almisbar BD: 0.2245.

The above coefficients show that STMs are attributed highest coefficients comparatively to BDs. The main reason for this is that EM algorithm penalizes models which assign zero probabilities to target-text words, and BDs will assign zero probabilities more often than STMs. Therefore this combination will usually advantage STMs than BDs even though the translations they propose may not be accurate.

Finally, the weight associated to each translation using linear combination is calculated with formula (5) (Section 3). Note also that after combination, if a query term is translated with several alternatives we keep at most four of them. This selection gives the best performance.

Table3. Performance (MAP) of classical models

Translation Model	TREC 2001	TREC 2002	Merged TREC 2001/2002
Monolingual IR	(0.33)	(0.28)	(0.31)
STM-Web	0.14 (42%)	0.04 (17%)	0.07 (25%)
STM-UN	0.11 (33%)	0.09 (34%)	0.10 (33%)
Ajeeb BD	0.27 (81%)	0.19 (70%)	0.22 (70%)
Almisbar BD	0.17 (51%)	0.16 (58%)	0.16 (54%)
LC	0.24 (72%)	0.20 (71%)	0.21 (67%)

Table 3 shows the performance of CLIR in terms of Mean Average Precision (MAP) using the four resources separately and combined linearly. We observe that the performance is quite different from one model to another. The low score recorded by the STM-Web is due to the small data set on which the STM is trained. 2 816 English-Arabic pairs of documents is not enough to build a reasonable STM. The other STM-UN trained on a large parallel corpora, produces slightly better results. Here, BDs present better performance than STMs because they provide multiple good translations to each query term. However, Almisbar results are not as good as those of Ajeeb because a lot of query terms are not covered by this BD. When combining all the resources, the performance is supposed to be better than with any individual resource because all query terms can be translated

[7] Arabic TREC collection. http://trec.nist.gov/

correctly by at least one resource. Results in table 3 do not confirm this assumption, especially for TREC 2001, where an individual resource (Ajeeb BD) performs better than the linear combination. This is due to the weak confidence coefficient attributed to this resource in the combination. Here is an example of English queries of TREC 2001: "What measures are being taken to develop tourism in Cairo?". The Arabic translation provided by TREC to the word "measures" is: "إجراءات". The table below shows the different translations provided by the four methods to this word.

Table4. Translation of "measures" with different methods

Translation model	Translation(s) of the word: "measures"
Ajeeb BD	تدبير 0.05 (measure), عيار 0.05 (caliber), قياس 0.05 (measurement), مقياس 0.05 (measurement), معيار 0.05 (standard), مكيال 0.05 (standard), ميزان 0.05 (balance)
Almisbar BD	إجراءات 0.05 (procedures), مقياس 0.03, قدر 0.03 (measurement), مقدار 0.03 (amount)
STM-UN	تدابير 0.69 (measures)
STM-Web	إجراءات 0.09
LC	إجراءات 0.029, مقياس 0.037, تدابير 0.61, قياس 0.020

We see clearly that translations with different resources are different. Some resources propose inappropriate translations such as "مكيال" or "ميزان". Even if two resources suggest the same translations, the weights are different. For this query, the linear combination produces better query translation terms than every resource taken alone: The most probable translations are selected from the combined list. However, this method is unable to attribute an appropriate weight to the best translation "إجراءات"; it is selected but ranked at third position with a weak weight. This example shows the limitation of the linear combination method: Even it selects the most probable translations it can not attribute appropriate weights to these translations.

5.2 CLIR with Confidence Measures (CM)

In these experiments, we select the four translations with the best confidences for each query term (as in LC). The following tables show the results:

Table5. Comparison of CLIR performance between linear combination and confidence measures

Collection	TREC 2001	TREC 2002	Merged TREC 2001/2002
MAP of LC	0.2426	0.2032	0.2163
MAP of CM	0.2775	0.2052	0.2290
Improvement rate of CM compared to LC	14.35 %	1 %	5.87 %

Table6. Precision at *n* retrieved documents with linear combination and confidence measures

Precision	TREC 2001		TREC 2002		Merged TREC 2001/2002	
	LC	CM	LC	CM	LC	CM
At 5 Docs	0.46	0.51	0.31	0.33	0.36	0.39
At 10 Docs	0.44	0.48	0.31	0.31	0.35	0.36
At 15 Docs	0.40	0.47	0.28	0.29	0.32	0.34
At 20 Docs	0.39	0.45	0.26	0.28	0.30	0.33
At 30 Docs	0.38	0.41	0.24	0.26	0.29	0.31
At 100 Docs	0.30	0.33	0.20	0.21	0.23	0.25
At 200 Docs	0.25	0.26	0.17	0.18	0.19	0.20
At 500 Docs	0.16	0.15	0.11	0.11	0.13	0.12
At 1000 Docs	0.10	0.09	0.07	0.07	0.08	0.08

In terms of mean average precision, we see clearly that the results using confidence measures are better than those obtained with the linear combination on both the two collections especially at lower levels of recall (Table 6). The improvement on the TREC 2001 collection (14.35 %) is greater than on the other collection (1 %) because the linear combination score on the former collection is weaker than the score obtained using a single resource (Ajeeb BD). The two-tailed t-test built with the results of table 6 shows that the improvement brought by confidence measures over linear combination is statistically significant at the level $P<0.05$. This improvement in CLIR performance is attributed to the ability of confidence measure to re-weight each translation candidate. The final set of translation words (and their probabilities) are more reasonable than in linear combination. For example, we get a large improvement in average precision for the TREC 2001 query "What measures are being taken to develop tourism in Cairo?", when translated using confidence measures. The query term "measures" is translated as follows by different methods:

Table7. Translation of "Measures" using LC and CM

Translation Model	Translation(s) of the query term "measures"
LC	قياس 0.029, إجراءات 0.037, مقياس 0.61, تدابير 0.0.20
CM	قياس 0.06, قدر 0.10, إجراءات 0.51

In this example, confidence measure has been able to increase the correct translation "إجراءات" to a higher level than the other incorrect or inappropriate ones. This example shows the potential advantage of confidence measures over linear combination: Linear combination assumes that all the suggested candidates are correct and it simply groups them together. On the contrary, the confidence model does not blindly trust all the translations. It tests their validity on new validation data. Thus, the translation candidates are rescored and filtered according to a more reliable

weight. In some examples, even if the confidence measure method proposes the same translations as the linear combination, the weights are readjusted. We do not claim that confidence measure is able to attribute accurate weights of importance to all translations, but the most likely translations are validated and rescored with higher weights. This strongly impacts the effectiveness of CLIR. Therefore, confidence measure provides a promising mechanism to select the most appropriate translations of a query.

In comparison with Ajeeb BD (Table 3), the retrieval effectiveness with confidence measures can still be improved slightly in all the cases. This shows that confidence measures are able to trust reliable translations for individual words, contrarily to the linear combination.

6. CONCLUSION

Most previous studies in CLIR used a linear combination of resources. This method is unable to combine correctly non homogeneous resources such as BDs and STMs. This naïve combination can keep noise proposed by the resources or attribute incorrect weights to translations. In this study, we examined the possibility of using a confidence measure technique for the query translation task. This method reconsiders each translation candidate proposed by different resources with respect to additional features. It is able to re-weight the translation candidate more radically than in linear combination. Our experiments show very encouraging results. We obtain an average improvement of 5.87 % compared to linear combination. This approach can be further improved on several aspects. For example, we can optimize this technique by identifying other informative features. Other techniques for computing confidence estimates can also be used in order to improve the performance of CLIR.

7. REFERENCES

[1] Al-Onaizan, Y., Curin, J., Jahr, M., Knight, K., Lafferty, J., Melamed, D., Och, F., Purdy, D., Smith, N., and Yarowsky, D. Statistical Machine Translation. Technical Report, CLSP/JHU 99 Workshop, Baltimore, MD, 1999.

[2] Blatz, J., Fitzgerald, E., Foster, G., Gandrabur, S., Goutte, C., Kulesza, A., Sanchis, A., and Ueffing, N. Confidence estimation for machine translation. Technical Report, CLSP/JHU 2003 Summer Workshop, Baltimore, 2003.

[3] Brown, P. F., Pietra, S. A., Pietra, V. J., and Mercer, R. L. The mathematics of statistical machine translation: Parameter estimation. Computational Linguistics, 19(2), 263–311, 1993.

[4] Fraser, A., Xu, J., and Weischedel, R. TREC 2002 Cross-lingual Retrieval at BBN. TREC11 conference, 2002.

[5] Gandrabur, S., and Foster, G. Confidence Estimation for Text Prediction. Proceedings of the Conference on Natural Language Learning (CoNLL 2003), Edmonton, May 2003.

[6] Hazen, T. J., Burianek, T., Polifroni, J., and Seneff, S. Recognition confidence scoring for use in speech understanding systems. Computer Speech and Language, Num. 16, pp. 49-67, 2002.

[7] Kadri, Y., and Nie, J. Y. Query translation for English-Arabic cross language information retrieval. Proceedings of the TALN conference, 2004.

[8] Kadri, Y., and Nie, J. Y. Effective stemming for Arabic information retrieval. The challenge of Arabic for NLP/MT Conference. The British Computer Society. London, UK, 2006.

[9] Nie, J. N., Simard, M., and Foster, G. Multilingual information retrieval based on parallel texts from the Web. In LNCS 2069, C. Peters editor, CLEF2000, pages 188-201, Lisbon 2000.

[10] Oard, D. W., and Diekema, A. Cross-Language Information Retrieval. In M. Williams (ed.), Annual review of Information science, 1998:223-256, 1998.

[11] Vogel, S., and Monson, C. Augmenting Manual Dictionaries for Statistical Machine Translation Systems. Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC), 2004.

[12] Xu, J., and Weischedel, R. Empirical studies on the impact of lexical resources on CLIR performance. Information processing & management, 41(3), 475-487, 2005.

[13] Zhai, C., and Lafferty, J. Model-based feedback in the language modeling approach to information retrieval. Tenth International Conference on Information and Knowledge Management (CIKM 2001), 2001.

[14] Zhai, C., and Lafferty, J. A study of smoothing methods for language models applied to ad hoc information retrieval. Proceedings of the ACM-SIGIR, 2001.