DOMCAT: A Bilingual Concordancer for Domain-Specific Computer Assisted Translation

Ming-Hong Bai[1,2] **Yu-Ming Hsieh**[1,2] **Keh-Jiann Chen**[1] **Jason S. Chang**[2]
1 Institute of Information Science, Academia Sinica, Taiwan
2 Department of Computer Science, National Tsing-Hua University, Taiwan
mhbai@sinica.edu.tw, morris@iis.sinica.edu.tw,
kchen@iis.sinica.edu.tw, jason.jschang@gmail.com

Abstract

In this paper, we propose a web-based bilingual concordancer, DOMCAT[1], for domain-specific computer assisted translation. Given a multi-word expression as a query, the system involves retrieving sentence pairs from a bilingual corpus, identifying translation equivalents of the query in the sentence pairs (translation spotting) and ranking the retrieved sentence pairs according to the relevance between the query and the translation equivalents. To provide high-precision translation spotting for domain-specific translation tasks, we exploited a normalized correlation method to spot the translation equivalents. To ranking the retrieved sentence pairs, we propose a correlation function modified from the Dice coefficient for assessing the correlation between the query and the translation equivalents. The performances of the translation spotting module and the ranking module are evaluated in terms of precision-recall measures and coverage rate respectively.

1 Introduction

A bilingual concordancer is a tool that can retrieve aligned sentence pairs in a parallel corpus whose source sentences contain the query and the translation equivalents of the query are identified in the target sentences. It helps not only on finding translation equivalents of the query but also presenting various contexts of occurrence. As a result, it is extremely useful for bilingual lexicographers, human translators and second language learners (Bowker and Barlow 2004; Bourdaillet et al., 2010; Gao 2011).

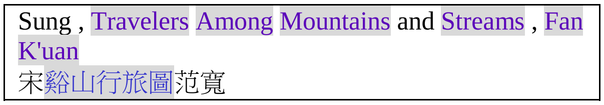
Identifying the translation equivalents, *translation spotting*, is the most challenging part of a bilingual concordancer. Recently, most of the existing bilingual concordancers spot translation equivalents in terms of word alignment-based method. (Jian et al., 2004; Callison-Burch et al., 2005; Bourdaillet et al., 2010). However, word alignment-based translation spotting has some drawbacks. First, aligning a rare (low frequency) term may encounter the *garbage collection effect* (Moore, 2004; Liang et al., 2006) that cause the term to align to many unrelated words. Second, the statistical word alignment model is not good at many-to-many alignment due to the fact that translation equivalents are not always correlated in lexical level. Unfortunately, the above effects will be intensified in a domain-specific concordancer because the queries are usually domain-specific terms, which are mostly multi-word low-frequency terms and semantically non-compositional terms.

Wu et al. (2003) employed a statistical association criterion to spot translation equivalents in their bilingual concordancer. The association-based criterion can avoid the above mentioned effects. However, it has other drawbacks in translation spotting task. First, it will encounter the *contextual effect* that causes the system incorrectly spot the translations of the strongly collocated context. Second, the association-based translation spotting tends to spot the *common subsequence* of a set of similar translations instead of the full translations. Figure 1 illustrates an example of *contextual effect*, in which 'Fan K'uan' is incorrectly spotted as part of the translation of the query term '谿山行旅圖' (Travelers Among Mountains and Streams), which is the name of the

[1] http://ckip.iis.sinica.edu.tw/DOMCAT/

Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pages 55–60,
Jeju, Republic of Korea, 8-14 July 2012. ©2012 Association for Computational Linguistics

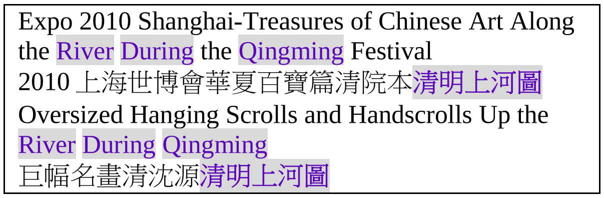
painting painted by ‘Fan K'uan/范寬’ since the painter’s name is strongly collocated with the name of the painting.



Sung , Travelers Among Mountains and Streams , Fan K'uan
宋谿山行旅圖范寬

Figure 1. ‘Fan K'uan’ may be incorrectly spotted as part of the translation of ‘谿山行旅圖’, if pure association method is applied.

Figure 2 illustrates an example of *common subsequence effect*, in which ‘清明上河圖’ (the River During the Qingming Festival/ Up the River During Qingming) has two similar translations as quoted, but the Dice coefficient tends to spot the *common subsequences* of the translations. (Function words are ignored in our translation spotting.)



Expo 2010 Shanghai-Treasures of Chinese Art Along the River During the Qingming Festival
2010 上海世博會華夏百寶篇清院本清明上河圖
Oversized Hanging Scrolls and Handscrolls Up the River During Qingming
巨幅名畫清沈源清明上河圖

Figure 2. The Dice coefficient tends to spot the common subsequences ‘River During Qingming’.

Bai et al. (2009) proposed a *normalized frequency* criterion to extract translation equivalents form sentence aligned parallel corpus. This criterion takes lexical-level contexture effect into account, so it can effectively resolve the above mentioned effect. But the goal of their method is to find most common translations instead of spotting translations, so the normalized frequency criterion tends to ignore rare translations.

In this paper, we propose a bilingual concordancer, DOMCAT, for computer assisted domain-specific term translation. To remedy the above mentioned effects, we extended the *normalized frequency* of Bai et al. (2009) to a *normalized correlation* criterion to spot translation equivalents. The *normalized correlation* inherits the characteristics of normalized frequency and is adjusted for spotting rare translations. These characteristics are especially important for a domain-specific bilingual concordancer to spot translation pairs of low-frequency and semantically non-compositional terms.

The remainder of this paper is organized as follows. Section 2 describes the DOMCAT system. In Section 3, we describe the evaluation of the DOMCAT system. Section 4 contains some concluding remarks.

2 The DOMCAT System

Given a query, the DOMCAT bilingual concordancer retrieves sentence pairs and spots translation equivalents by the following steps:

1. Retrieve the sentence pairs whose source sentences contain the query term.
2. Extract translation candidate words from the retrieved sentence pairs by the normalized correlation criterion.
3. Spot the candidate words for each target sentence and rank the sentences by normalized the Dice coefficient criterion.

In step 1, the query term can be a single word, a phrase, a gapped sequence and even a regular expression. The parallel corpus is indexed by the suffix array to efficiently retrieve the sentences.

The step 2 and step 3 are more complicated and will be described from Section 2.1 to Section 2.3.

2.1 Extract Translation Candidate Words

After the queried sentence pairs retrieved from the parallel corpus, we can extract translation candidate words from the sentence pairs. We compute the *local normalized correlation* with respect to the query term for each word *e* in each target sentence. The *local normalized correlation* is defined as follows:

$$lnc(e;\mathbf{q},\mathbf{e},\mathbf{f}) = \frac{\sum_{\forall f_i \in \mathbf{q}} p(e\,|\,f_i)+\Delta\,|\,\mathbf{q}\,|}{\sum_{\forall f_j \in \mathbf{f}} p(e\,|\,f_j)+\Delta\,|\,\mathbf{f}\,|} \qquad (1)$$

where **q** denotes the query term, **f** denotes the source sentence and **e** denotes the target sentence, Δ is a small smoothing factor. The probability *p*(*e*|*f*) is the word translation probability derived from the entire parallel corpus by IBM Model 1 (Brown et al., 1993). The sense of *local normalized correlation* of *e* can be interpreted as the probability of word *e* being part of translation of the query term **q** under the condition of sentence pair (**e, f**).

Once the local normalized correlation is computed for each word in retrieved sentences, we compute the normalized correlation on the retrieved sentences. The normalized correlation is the average of all *lnc* values and defined as follows:

$$nc(e;\mathbf{q}) = \frac{1}{n}\sum_{i=1}^{n} lnc(e;\mathbf{q},\mathbf{e}^{(i)},\mathbf{f}^{(i)}) \qquad (2)$$

where *n* is the number of retrieved sentence pairs.

After the *nc* values for the words of the retrieved target sentences are computed, we can obtain a translation candidate list by filtering out the words with lower *nc* values.

To compare with the association-based method, we also sorted the word list by the Dice coefficient defined as follows:

$$dice(e,\mathbf{q}) = \frac{2\,freq(e,\mathbf{q})}{freq(e) + freq(\mathbf{q})} \qquad (3)$$

where *freq* is frequency function which computes frequencies from the parallel corpus.

Candidate words	NC
mountain	0.676
stream	0.442
traveler	0.374
among	0.363
sung	0.095
k'uan	0.090

Figure 3(a). Candidate words sorted by *nc* values.

Candidate words	Dice
traveler	0.385
reduced	0.176
stream	0.128
k'uan	0.121
fan	0.082
among	0.049
mountain	0.035

Figure 3(b). Candidate words sorted by Dice coefficient values.

Figure 3(a) and (b) illustrate examples of translation candidate words of the query term '谿山行旅圖' (Travelers Among Mountains and Streams) sorted by the *nc* values, *NC*, and the Dice coefficients respectively. The result shows that the normalized correlation separated the related words from unrelated words much better than the Dice coefficient.

The rationale behind the normalized correlation is that the *nc* value is the strength of word *e* generated by the query compared to that of generated by the whole sentence. As a result, the normalized correlation can easily separate the words generated by the query term from the words generated by the context. On the contrary, the Dice coefficient counts the frequency of a co-occurred word without considering the fact that it could be generated by the strongly collocated context.

2.2 Translation Spotting

Once we have a translation candidate list and respective *nc* values, we can spot the translation equivalents by the following spotting algorithm. For each target sentence, first, spot the word with highest *nc* value. Then extend the spotted sequence to the neighbors of the word by checking their *nc* values of neighbor words but skipping function words. If the *nc* value is greater than a threshold θ, add the word into spotted sequence. Repeat the extending process until no word can be added to the spotted sequence.

The following is the pseudo-code for the algorithm:

```
S is the target sentence
H is the spotted word sequence
$\theta$ is the threshold of translation candidate words

Initialize:
    H←⍰
    $e_{max}$←S[0]
Foreach $e_i$ in S:
    If $nc(e_i) > nc(e_{max})$:
        $e_{max} \leftarrow e_i$
If $nc(e_{max}) > \theta$:
    add $e_{max}$ to H
Repeat until no word add to H
    $e_j$←left neighbor of H
    If $nc(e_j) > \theta$:
        add $e_j$ to H
    $e_k$←right neighbor of H
    If $nc(e_k) > \theta$:
        add $e_k$ to H
```

Figure 4: Pseudo-code of translation spotting process.

2.3 Ranking

The ranking mechanism of a bilingual concordancer is used to provide the most related translation of the query on the top of the outputs for the user. So, an association metric is needed to evaluate the relations between the query and the spotted translations. The Dice coefficient is a widely used measure for assessing the association strength between a multi-word expression and its translation candidates. (Kupiec, 1993; Smadja et al., 1996; Kitamura and Matsumoto, 1996; Yamamoto and Matsumoto, 2000; Melamed, 2001) The following is the definition of the Dice coefficient:

$$dice(\mathbf{t},\mathbf{q}) = \frac{2\,freq(\mathbf{t},\mathbf{q})}{freq(\mathbf{t}) + freq(\mathbf{q})} \qquad (4)$$

where **q** denotes a multi-word expression to be translated, **t** denotes a translation candidate of **q**. However, the Dice coefficient has the *common subsequence effect* (as mentioned in Section 1) due to the fact that the co-occurrence frequency of the common subsequence is usually larger than that of the full translation; hence, the Dice coefficient tends to choose the common subsequence.

To remedy the common subsequence effect, we introduce a *normalized frequency* for a spotted sequence defined as follows:

$$nf\,(\mathbf{t},\mathbf{q}) = \sum_{i=1}^{n} lnf\,(\mathbf{t};\mathbf{q},\mathbf{e}^{(i)},\mathbf{f}^{(i)}) \qquad (5)$$

where *lnf* is a function which compute normalized frequency locally in each sentence. The following is the definition of *lnf*:

$$lnf\,(\mathbf{t};\mathbf{q},\mathbf{e},\mathbf{f}) = \prod_{\forall e \in \mathbf{H}-\mathbf{t}} (1 - lnc(e;\mathbf{q},\mathbf{e},\mathbf{f})) \qquad (6)$$

where ***H*** is the spotted sequence of the sentence pair (**e**,**f**), ***H-t*** are the words in **H** but not in **t**. The rationale behind *lnf* function is that: when counting the local frequency of **t** in a sentence pair, if **t** is a subsequence of **H**, then the count of **t** should be reasonably reduced by considering the strength of the correlation between the words in **H-t** and the query.

Then, we modify the Dice coefficient by replacing the co-occurrence frequency with normalized frequency as follows:

$$nf\_dice(\mathbf{t},\mathbf{q}) = \frac{2nf\,(\mathbf{t},\mathbf{q})}{freq(\mathbf{t}) + freq(\mathbf{q})} \qquad (7)$$

The new scoring function, *nf_dice*(**t**,**q**), is exploited as our criterion for assessing the association strength between the query and the spotted sequences.

3 Experimental Results

3.1 Experimental Setting

We use the Chinese/English web pages of the National Palace Museum [2] as our underlying parallel corpus. It contains about 30,000 sentences in each language. We exploited the Champollion Toolkit (Ma et al., 2006) to align the sentence pairs. The English sentences are tokenized and lemmatized by using the NLTK (Bird and Loper, 2004) and the Chinese sentences are segmented by the CKIP Chinese segmenter (Ma and Chen, 2003).

To evaluate the performance of the translation spotting, we selected 12 domain-specific terms to query the concordancer. Then, the returned spotted translation equivalents are evaluated against a manually annotated gold standard in terms of recall and precision metrics. We also build two different translation spotting modules by using the GIZA++ toolkit (Och and Ney, 2000) with the intersection/union of the bidirectional word alignment as baseline systems.

To evaluate the performance of the ranking criterion, we compiled a reference translation set for each query by collecting the manually annotated translation spotting set and selecting 1 to 3 frequently used translations. Then, the outputs of each query are ranked by the *nf_dice* function and evaluated against the reference translation set. We also compared the ranking performance with the Dice coefficient.

3.2 Evaluation of Translation Spotting

We evaluate the translation spotting in terms of the Recall and Precision metrics defined as follows:

[2] http://www.npm.gov.tw

$$Recall = \frac{\sum_{i=1}^{n} | H_g^{(i)} \cap H^{(i)} |}{\sum_{i=1}^{n} | H_g^{(i)} |} \tag{8}$$

$$Precision = \frac{\sum_{i=1}^{n} | H_g^{(i)} \cap H^{(i)} |}{\sum_{i=1}^{n} | H^{(i)} |} \tag{9}$$

where i denotes the index of the retrieved sentence, $H^{(i)}$ is the spotted sequences of the ith sentence returned by the concordancer, and $H_g^{(i)}$ is the gold standard spotted sequences of the ith sentence. Table 1 shows the evaluation of translation spotting for normalized correlation, *NC*, compared with the intersection and union of GIZA++ word alignment. The F-score of the normalized correlation is much higher than that of the word alignment methods. It is noteworthy that the normalized correlation increased the recall rate without losing the precision rate. This may indicate that the normalized correlation can effectively conquer the drawbacks of the word alignment-based translation spotting and the association-based translation spotting mentioned in Section 1.

	Recall	Precision	F-score
Intersection	0.4026	0.9498	0.5656
Union	0.7061	0.9217	0.7996
NC	0.8579	0.9318	0.8933

Table 1. Evaluation of the translation spotting queried by 12 domain-specific terms.

We also evaluate the queried results of each term individually (as shown in Table 2). As it shows, the normalized correlation is quite stable for translation spotting.

Query terms	GIZA Intersection			GIZA Union			NC		
	R	P	F	R	P	F	R	P	F
毛公鼎 (Maogong cauldron)	0.27	0.86	0.41	0.87	0.74	0.80	0.92	0.97	**0.94**
翠玉白菜(Jadeite cabbage)	0.48	1.00	0.65	1.00	0.88	0.94	0.98	0.98	**0.98**
谿山行旅圖(Travelers Among Mountains and Streams)	0.28	0.75	0.41	1.00	0.68	0.81	0.94	0.91	**0.92**
清明上河圖(Up the River During Qingming)	0.22	0.93	0.35	0.97	0.83	0.89	0.99	0.91	**0.95**
景德鎮(Ching-te-chen)	0.50	0.87	0.63	0.73	0.31	0.44	1.00	0.69	**0.82**
瓷器(porcelain)	0.53	0.99	0.69	0.93	0.64	0.76	0.78	0.96	**0.86**
霽青(cobalt blue glaze)	0.12	1.00	0.21	0.85	0.58	0.69	0.94	0.86	**0.90**
銘文(inscription)	0.20	0.89	0.32	0.71	0.34	0.46	0.88	0.95	**0.91**
三友百禽(Three Friends and a Hundred Birds)	0.58	0.99	0.73	1.00	0.97	**0.99**	1.00	0.72	0.84
狂草(wild cursive script)	0.42	1.00	0.59	0.63	0.80	0.71	0.84	1.00	**0.91**
蘭亭序(Preface to the Orchid Pavilion Gathering)	0.33	0.75	0.46	0.56	0.50	0.53	0.78	1.00	**0.88**
後赤壁賦(Latter Odes to the Red Cliff)	0.19	0.50	0.27	0.75	0.46	0.57	0.94	0.88	**0.91**

Table 2. Evaluation of the translation spotting for each term

3.3 Evaluation of Ranking

To evaluate the performance of a ranking function, we ranked the retrieved sentences of the queries by the function. Then, the top-n sentences of the output are evaluated in terms of the coverage rate defined as follows:

$$\text{coverage} = \frac{\#\text{ of queries can find a translation in top-n}}{\#\text{ of queries}} \tag{10}$$

The meaning of the coverage rate can be interpreted as: how many percent of the query can find an acceptable translation in the top-n results. We use the reference translations, as described in Section 3.1, as acceptable translation set for each query of our experiment. Table 3 shows the coverage rate of the *nf_dice* function compared with the Dice coefficient. As it shows, in the outputs ranked by the Dice coefficient, uses usually have to look up more than 3 sentences to find an acceptable translation; while in the outputs ranked by the *nf_dice* function, users can find an acceptable translation in top-2 sentences.

	dice	nf_dice
top-1	0.42	0.92
top-2	0.75	1.00
top-3	0.92	1.00

Table 3. Evaluation of the ranking criteria.

4 Conclusion and Future Works

In this paper, we proposed a bilingual concordancer, DOMCAT, designed as a domain-specific computer assisted translation tool. We exploited a normalized correlation which incorporate lexical level information into association-based method that effectively avoid the drawbacks of the word alignment-based translation spotting as well as the association-based translation spotting.

In the future, it would be interesting to extend the parallel corpus to the internet to retrieve more rich data for the computer assisted translation.

References

Bai, Ming-Hong, Jia-Ming You, Keh-Jiann Chen, Jason S. Chang. 2009. Acquiring Translation Equivalences of Multiword Expressions by Normalized Correlation Frequencies. In *Proceedings of EMNLP*, pages 478-486.

Bird, Steven and Edward Loper. 2004. NLTK: The Natural Language Toolkit. In *Proceedings of ACL*, pages 214-217.

Bourdaillet, Julien, Stéphane Huet, Philippe Langlais and Guy Lapalme. 2010. TRANSSEARCH: from a bilingual concordancer to a translation finder. *Machine Translation*, 24(3-4): 241–271.

Bowker, Lynne, Michael Barlow. 2004. Bilingual concordancers and translation memories: A comparative evaluation. In *Proceedings of the Second International Workshop on Language Resources for Translation Work, Research and Training* , pages. 52-61.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, Robert L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263-311.

Callison-Burch, Chris, Colin Bannard and Josh Schroeder. 2005. A Compact Data Structure for Searchable Translation Memories. In *Proceedings of EAMT*.

Gao, Zhao-Ming. 2011. Exploring the effects and use of a Chinese–English parallel concordancer. *Computer-Assisted Language Learning* 24.3 (July 2011): 255-275.

Jian, Jia-Yan, Yu-Chia Chang and Jason S. Chang. 2004. TANGO: Bilingual Collocational Concordancer. In *Proceedings of ACL*, pages 166-169.

Kitamura, Mihoko and Yuji Matsumoto. 1996. Automatic Extraction of Word Sequence Correspondences in Parallel Corpora. In *Proceedings of WVLC-4* pages 79-87.

Kupiec, Julian. 1993. An Algorithm for Finding Noun Phrase Correspondences in Bilingual Corpora. In *Proceedings of ACL*, pages 17-22.

Liang, Percy, Ben Taskar, Dan Klein. 2006. Alignment by Agreement. In *Proceedings of HLT-NAACL 2006*, pages 104-111, New York, USA.

Ma, Wei-Yun and Keh-Jiann Chen. 2003. Introduction to CKIP Chinese word segmentation system for the first international Chinese word segmentation bakeoff. In *Proceedings of the second SIGHAN workshop on Chinese language processing*, pages 168-171.

Ma, Xiaoyi. 2006. Champollion: A Robust Parallel Text Sentence Aligner. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation.*

Melamed, Ilya Dan. 2001. Empirical Methods for Exploiting parallel Texts. *MIT press*.

Moore, Robert C. 2004. Improving IBM Word-Alignment Model 1. In *Proceedings of ACL*, pages 519-526, Barcelona, Spain.

Och, Franz J., Hermann Ney., 2000, Improved Statistical Alignment Models, In *Proceedings of ACL*, pages 440-447. Hong Kong.

Smadja, Frank, Kathleen R. McKeown, and Vasileios Hatzivassiloglou. 1996. Translating Collocations for Bilingual Lexicons: A Statistical Approach. *Computational Linguistics*, 22(1):1-38.

Wu, Jian-Cheng, Kevin C. Yeh, Thomas C. Chuang, Wen-Chi Shei, Jason S. Chang. 2003. TotalRecall: A Bilingual Concordance for Computer Assisted Translation and Language Learning. In *Proceedings of ACL*, pages 201-204.

Yamamoto, Kaoru, Yuji Matsumoto. 2000. Acquisition of Phrase-level Bilingual Correspondence using Dependency Structure. In *Proceedings of COLING*, pages 933-939.