An EM Algorithm for Context-Based Searching and Disambiguation with Application to Synonym Term Alignment *

Jing-Shin Chang and Shih-Jay Chiou

Department of Computer Science and Information Engineering
National Chi Nan University
1, Univ. Road, Puli, Nantou 545, Taiwan, ROC.
jshin@csie.ncnu.edu.tw, s96321502@ncnu.edu.tw

Abstract. A statistical context-based searching model and an unsupervised EM algorithm are proposed to resolve the large class of searching problems that require left and right contexts for disambiguation, in which the contexts can be synonyms. The searching problem is modeled as a machine translation problem in which pieces of contexts are accumulated to enforce the translation probability between a search result and the source query. This model is applied to the term alignment problem between traditional and simplified Chinese synonymous terms. In comparison with previous works on the same task, the EM algorithm for context-based searching and disambiguation significantly improves the term alignment accuracy by 2~48%, for technical, transliteration and common terms. The alignment accuracy ranges from 47~85% in different domains.

Keywords: context-based searching, disambiguation, simplified-traditional Chinese term alignment, lexical semantics, statistical machine translation

1 Motivation for Context-Based Searching and Disambiguation

1.1 Ambiguity and Translation Equivalence in Searching

Searching is one of the largest application type on the Web today. It ranges from document search to media search, mostly based on contextual texts around the searching targets. Searching strategies for various NLP resources are also developed now by the natural language processing (NLP) community for utilizing the web as an extremely huge text corpus. Therefore, accurate searching results will provide useful research resources, such as translation equivalents between two language or synonymous terms.

In its native form, a query is given to a search engine and the search engine returns some relevant results as output. If all the query terms (words or phrases) and their searching targets in text materials are unambiguous, then searching can be conducted very precisely by using exact pattern matching. Unfortunately, both "query" terms and "results" might be highly ambiguous. The returned results, therefore, might not be relevant to the user intention. As a result, one needs to resolve various kinds of ambiguity using contextual information in a longer and well formulated query or using contexts around the searching target.

Ambiguity comes from several sources. First of all, one query and their targets may have multiple word senses. One-to-one exact matching is thus impossible. Secondly, the relevant searching objects may not be described using the same terms as the user query. Instead, they may be described in terms of synonymous terms or translation equivalents in another language. In this case, the query may have to be expanded by its synonyms or translation to match relevant searching targets. All these problems require contextual information in the queries (in

* Acknowledgements: This work is partially supported by the National Science Council (NSC), Taiwan, Republic of China (ROC), under the contract NSC 96-2221-E-260-022-.

order to specify user intention more specifically) or in the documents (in order to match user intention more closely) for disambiguation. Unfortunately, the contexts themselves could also be ambiguous and could be synonyms or translation equivalent of other terms. A third problem therefore arisen from the multiple senses and multiple equivalent forms of the contexts themselves. In this work we will propose an EM algorithm to resolve the sense ambiguity and translation equivalent problems associated not only with the query and searching targets but also with the contexts in a unified framework.

1.2 Context-Based Searching Problem

For simplicity, we assume that contexts are unambiguous in the first place. In general, a query is likely to be relevant to a candidate searching object if they share the same contexts in many instances. In order to resolve the various searching problems that requires left and/or right contexts of the key query term and/or the target searching objects, we can formulate a context-based searching (or disambiguation) problem as a 7-tuple <S, T, Ls, Rs, Lt, Rt, A>. In the above vector, S is the key query term, or the source object, that must be submitted to a search engine; T is the candidate target object to search for; Ls and Rs are the left and right contexts of S; and, Lt and Rt are the left and right contexts of T, respectively. Note that Ls and Rs may formulate S more precisely; and Lt and Rt may serve to disambiguate T in the particular context. By distinguishing the contexts into left and right contexts, we implicitly imply that contexts are "directional". If this constraint is not absolutely necessary, we can introduce an alignment vector, A, which serves to re-order the left/right contexts in some order-free ("unidirectional") manner so any contextual terms can be matched against any other contextual terms to provide evidences whether S and T are the right matching pair.

The goal of a context-based search (or disambiguation) process is then to find the most relevant search result(s), T, given a main source query term, S, with the help of L/R contexts. Intuitively, S and T tend to be a relevant query-answer pair if many contexts are "matched". The target object, T*, with highest matching score (or probability) will be the most possible target that S is referring to in the contexts of <Ls, Rs>.

The degree of matching can be measured in terms of different "matching strength" or "matching score" contributed by the contexts. Normally, *exact string match* between two terms in S/T or L/R contexts, such as "the Big Apple" *vs*. "the Big Apple", has the strongest match. But it is least robust since S/T/L/R might be described in terms of other synonymous form. *Partial or fuzzy match*, like "Big Apple" *vs*. "the Big Apple", provides some flexibility for matching. But it may also introduce noise such as matching "the Big Apple" against "Big Apple Pie".

The most robust and flexible way for matching S/T and L/R contexts might be to assign a higher matching score to a term pair if they are known to be synonyms or highly related terms in the ontology. For instance, it is desirable to know that "the Big Apple" and "N.Y." match each other in some contexts, and "happiness" (noun) matches "joyful" (adjective) to some extents. We will refer this level of matching as *synonymous match*, to distinguish it from exact (or partial) string match. Since the contexts themselves could be ambiguous or synonym of each other, a generic model capable of matching context at the synonymous level quantitatively will provide the most robust results for context-based matching.

In this paper, an EM algorithm, capable of searching relevant S-T pair at the synonymous matching level, is proposed. The searching problem is transformed into a statistical machine translation (SMT) problem in which S is translated correctly into T with the help of contexts around S and T for correct disambiguation. The translation probability between an S-T pair is accumulated from various contextual windows around S and T, each being assigned an individual window-wise translation score; the term-wise translation probabilities can be iteratively estimated using an unsupervised EM algorithm. Synonymous term pairs will gain higher and higher translation probabilities through the EM training, even though they may not look like synonyms or a highly relevant pair at first. This thus makes synonymous level matching possible. It is the power of synonymous level matching that makes such a model

attractive. In the end, the best T with the highest translation probability will be selected as the best translation or searching object.

1.3 Macroscopic Distributional Similarity vs. Microscopic Contextual Similarity

One popular way, in the information retrieval and NLP communities, to use contextual information for searching or disambiguation is to characterize the source query S and target object T in terms of the distribution of important contextual terms around them. The "distributional similarity", in terms of vector distance or KL distance of the two distributions, is then used to see if S and T are highly relevant (Lee, 1999). Unlike other distributional semantic similarity measures, the proposed EM model uses finer-grain, microscopic *local contexts* to enforce the evidence of contextual similarity. Furthermore, different contexts are weighted differently and *near contexts* prevail. Therefore, it is expected to provide better resolution for resolving sense ambiguity than those macroscopic distributional similarity measures.

For example, with a distributional similarity measure, the three syntactic patterns: "the color X-1", "color for the X-2" and "the X-3 color" will be treated the same, since the important contextual term "color" appears once identically in all these three contextual windows for the three unknown terms X-1, X-2 and X-3. This will suggest that X-1, X-2 and X-3 might refer to the same object in terms of distributional similarity. However, it is not hard to guess that X-1 is more likely to be a "picture" than a "car" or the "Kodak". On the other hand, X-2 and X-3 might be better searching targets for "car" and "Kodak", respectively. The proposed EM searching and disambiguation model will weight these syntactic patterns differently and accumulate many pieces of such syntactic evidences for better disambiguation.

To formulate this general context-based searching model and demonstrate its disambiguation capability, the following sections will use the term alignment problem between simplified Chinese (SC) terms and traditional Chinese (TC) terms, which was first raised in (Chang and Kung, 2007), as an example and extend it to the current EM framework for context-based searching. The improvement over this prior work is then demonstrated. The term alignment problem is to find the most likely synonymous term for a SC-specific term, such as "激光" from a list of TC-specific terms, such as "雷射", or *vice versa*. These terms can not be equated simply by exact pattern matching. And, the two variant forms with similar meanings will result in similar difficulties as cross-language searching problems. In this application, the source query S simply refers to an SC-specific term, and the searching target T is a candidate TC-specific term. And we have many contextual windows, some look like:

Ls-S-Rs: "一部 激光 打印機"

Lt-T-Rt: "一台 雷射 印表機"

which suggest that "激光" and "雷射" are likely to be synonymous terms since their left contexts "一部" and "一台" and right contexts "打印機" and "印表機" are known (gradually through EM training) to be equivalent with high degree of certain. Of course, the same model can be applied to other applications, such as media search, with minor extension.

Since the current EM framework is extended from the work in (Chang and Kung, 2007), the following sections will follow most logic of this original work. However, the original work relies on *exact* match against the contexts; it therefore cannot handle the situations where contexts are ambiguous or equivalent terms of others. We therefore will highlight the incremental adaptation which makes synonymous matching, instead of exact match, possible through the EM training.

2 Context-Based EM Algorithm for Aligning Simplified-Traditional Chinese

2.1 General EM-Based SMT Model for Term Searching

Formally, the SC-TC term alignment problem can be formulated as finding the most likely translation equivalent for SC-specific terms from the set of candidate TC-specific terms, or *vice versa*. Searching or aligning equivalent TC (Traditional Chinese) term for a particular SC

(Simplified Chinese) term can thus be modeled as a special Chinese-to-Chinese Statistical Machine Translation (C2C SMT) problem. To find the best traditional Chinese term, 't', corresponding to a simplified Chinese term 's' in the simplified Chinese lexicon, D_S, is then equivalent to finding the one with the highest translation probability, $P(t \mid s)$, among those terms in the traditional-specific dictionary D_T. That is,

$$\begin{aligned} t^* &= \arg\max_{t \in D_T} P(t \mid s) \qquad \forall s \in D_S \\ &= \arg\max_{t \in D_T} P(t, s) \end{aligned} \tag{1}$$

In general context-based searching problems, we can simply map 's' and 't' to the source query and target objects, respectively. Unlike traditional SMT (Brown et al., 1993) or alignment models (Och and Ney, 2000), however, general context-based searching problems will not have a parallel corpus for training. Instead, we have to acquire a large number of contextual windows which include the S-T pairs from text corpora or web pages to ensure that only right term pairs match each other. To introduce contextual information to the model, the contexts can be implemented as some hidden contextual variables. And the total translation probability can be estimated or marginalized by summing the context-aware translation probability, such as Pr(<一部, 數位, 相機>, <一部, 數碼, 相機>).

Intuitively, a simplified Chinese term (s) can be characterized by its left context l_s and right context r_s, and hence a contextual window $\langle l_s, s, r_s \rangle$. The same is true to representing a traditional Chinese term, t, at some particular context, with $\langle l_t, t, r_t \rangle$, where $\langle l_t, r_t \rangle$ are its left/right neighbors. If a sub-window $\langle l_s, s, r_s \rangle$ is "similar" to the triple $\langle l_t, t, r_t \rangle$, then s and t are likely to be translation equivalent of each other. In other words, Equation (1) can be modeled as: (Chang and Kung, 2007)

$$\begin{aligned} P(t, s) &= \sum_{\substack{\langle l_t, r_t \rangle : \langle l_t, t, r_t \rangle \in T_T \\ \langle l_s, r_s \rangle : \langle l_s, s, r_s \rangle \in T_S}} P\left(\langle l_t, t, r_t \rangle, \langle l_s, s, r_s \rangle\right) \\ t^* &= \arg\max_{t \in D_T} \sum_{\substack{\langle l_t, r_t \rangle : \langle l_t, t, r_t \rangle \in T_T \\ \langle l_s, r_s \rangle : \langle l_s, s, r_s \rangle \in T_S}} P\left(\langle l_t, t, r_t \rangle, \langle l_s, s, r_s \rangle\right) \end{aligned} \tag{2}$$

where T_T and T_S are the text corpora or web pages where s and t appear. Normally, the degree of context "similarity" between $\langle l_s, s, r_s \rangle$ and $\langle l_t, t, r_t \rangle$ is easier to estimate than judging the degree of "equivalence" between s and t directly. For example, if $l_s = l_t$ then the similarity of the two triples will be increased; if, in addition, $r_s = r_t$, then the "similarity" will be further enhanced.

With the above formulation, the complicated context-free s-to-t translation problem can be divided into a large number of easier context-dependent translation sub-problems. In other words, the probability $P(t, s)$ can be contributed additively by each $\langle l_s, s, r_s \rangle$ and $\langle l_t, t, r_t \rangle$ pair; those pairs with higher similarity will contribute more confidence than those that are unrelated. With such a context-based formulation, one may hopefully estimate $P(t, s)$ more reliably with more contextual information.

Sometimes, the word orders of the triple pairs may not be important in measuring the equivalence relationship. For instance, the different contexts like '一部 數碼 相機' and '數位 相機 一部' may still suggest that '數碼' and '數位' are equivalent. For languages with free word order, Equation (2) can further be expressed as: (Chang and Kung, 2007)

$$t^* = \arg\max_{t \in D_T} \sum_{\substack{\langle l_t, r_t \rangle : \langle l_t, t, r_t \rangle \in T_T \\ \langle l_s, r_s \rangle : \langle l_s, s, r_s \rangle \in T_S}} \sum_A P\left(A, \langle l_t, t, r_t \rangle, \langle l_s, s, r_s \rangle\right)$$

$$= \arg\max_{t \in D_T} \sum_{\substack{\langle t_{-1}, t_1 \rangle : \langle t_{-1}, t_0, t_1 \rangle \in T_T \\ \langle s_{-1}, s_1 \rangle : \langle s_{-1}, s_0, s_1 \rangle \in T_S}} \sum_{a_{-1}, a_0, a_1} P\left(A, \langle t_{-1}, t_0, t_1 \rangle, \langle s_{-1}, s_0, s_1 \rangle\right) \quad (3)$$

$$A \equiv \langle a_{-1}, a_0, a_1 \rangle, \langle l_t, t, r_t \rangle \equiv \langle t_{-1}, t_0, t_1 \rangle, \langle l_s, s, r_s \rangle \equiv \langle s_{-1}, s_0, s_1 \rangle$$

where A $= \langle a_{-1}, a_0, a_1 \rangle$ is an alignment vector associated with the simplified Chinese terms in the triple, such that $a_j = i$ if and only if s_j and t_{a_j} (i.e., t_i) are potential translation pair. For simplicity, the left/right terms are re-indexed in the second equality with the subscripts –1 and 1 respectively, and the central terms in focus are indexed with 0 in Equation (3).

To simplify the development of an EM algorithm, the alignment probability is assumed to be the product of the probabilities of each individual term pair.

$$\begin{aligned} & P\left(A, \langle l_t, t, r_t \rangle, \langle l_s, s, r_s \rangle\right) \\ &= P\left(A, \langle t_{-1}, t_0, t_1 \rangle, \langle s_{-1}, s_0, s_1 \rangle\right) \qquad (4) \\ &\equiv \prod_j P\left(t_{a_j}, s_j\right) \end{aligned}$$

More complicated models for the above alignment probability can be exploited though. For now, the simplicity of such an assumption will be taken throughout the training process.

2.2 An EM Algorithm for Estimating Alignment Scores

The alignment probability and other parameters can be trained using an EM algorithm (Dempster et al., 1977) as follows. Supposed that, for each $\langle s_{-1}, s_0, s_1 \rangle$ and $\langle t_{-1}, t_0, t_1 \rangle$ pair, $P\left(A, \langle t_{-1}, t_0, t_1 \rangle, \langle s_{-1}, s_0, s_1 \rangle\right)$ is properly initialized for any alignment pattern, A, then the expected counts $\hat{c}(t, s)$, of the E-Step, for each translation pair can be estimated as

$$\hat{c}(t, s) = \sum_{\langle t_{-1}, t_0, t_1 \rangle} \sum_{\langle s_{-1}, s_0, s_1 \rangle} \sum_A \sum_j P\left(A, \langle t_{-1}, t_0, t_1 \rangle, \langle s_{-1}, s_0, s_1 \rangle\right) \delta\left(t = t_{a_j}\right) \delta\left(s = s_j\right) \quad (5)$$

where $\delta(e) = 1$ if event e is true, $\delta(e) = 0$ if otherwise.

The expected counts, in turn, can be used, in the M-Step, to estimate $P(t, s)$ as

$$\hat{P}(t, s) = \frac{\hat{c}(t, s)}{\sum_{t,s} \hat{c}(t, s)}, \qquad (6)$$

and $P\left(A, \langle t_{-1}, t_0, t_1 \rangle, \langle s_{-1}, s_0, s_1 \rangle\right)$ can be re-estimated according to Equation (4). The training process then repeat itself until it converges (or a maximum number of training iteration is reached.) For a general context-base searching problem, the process to generate the contextual windows may be a bit different, depending on the nature of the application. Note that, in (Chang and Kung, 2007),

$$\begin{aligned} & P\left(A, \langle t_{-1}, t_0, t_1 \rangle, \langle s_{-1}, s_0, s_1 \rangle\right) \equiv K \sum_{j=-1,1} \delta\left(s_j = t_{a_j}\right) \\ & P(t, s) \equiv K \hat{c}(s, t) \qquad (7) \\ & \hat{c}(s, t) = \sum_{\langle l_t, r_t \rangle, \langle l_s, r_s \rangle} \sum_A \sum_j \delta\left(s_j = t_{a_j}\right) \end{aligned}$$

which means that the expected count will be incremented only when two aligned terms are the same, and each exact match contributes a constant probability K to the alignment probability. On the other hand, the current EM algorithm allows terms to be matched in a probabilistic sense; synonyms or translation equivalents will acquire a fractional expected count in proportional to the alignment probability. Therefore, the contexts themselves can be ambiguous or in an variant form; this does not prevent two terms to be matched probabilistically through the EM training. The synonymous matching is thus possible with the current EM algorithm.

3 Experiment Results and Discussion

3.1 Data Sources

This work uses 346 well-known simplified-traditional Chinese term pairs and their contextual windows collected by (Chang and Kung, 2007). The data come from three different domains. D1, D2 and D3 will represent the "technical terms", "transliterations", and "general terms", respectively, throughout this paper. There are 136, 82, and 128 term pairs, respectively, for domains D1~D3.

According to (Chang and Kung, 2007), the SC-specific terms and TC-specific terms are submitted to the Google search engine (http://www.google.com/), and one left word and one right word together with the central SC/TC term form a 3-term windows. All the snippets retuned from Google are word segmented (Chiang et al., 1992) using the union of the simplified and the traditional Chinese vocabularies derived from the first SIGHAN word segmentation bakeoff corpora (Sproat and Emerson, 2003). To reduce the effects of data sparseness, contextual windows that occur only once are removed from the training data. The evaluation of the context-based EM algorithm is given in the following sections.

3.2 Improved Contextual Similarity with the EM Algorithm

The EM algorithm is used in the current work to estimate the translation probability of each S-T pair, hoping that each correct alignment pair gets higher and higher translation probability iteration by iteration. Through the EM training process, it is also hoped that more and more correct alignment pairs will be identified. The following table lists some correct S-T term pairs in the technical domain and their log-scaled translation probabilities (log P(s, t)) in the first few iterations. The shaded cells indicate the translation probabilities at which the S-T pair in the first column is correctly identified as the most likely translation equivalent. For instance, '激光' and '雷射' were initially not recognized as equivalent terms, but are correctly matched since iteration-3.

As expected, the translation probabilities of correct alignment pairs get higher and higher scores with the help of the EM training process. Drawing the log P(s,t) curves as a function of the iteration numbers will show 3 monotonically increasing curves. These curves demonstrate that the context-based EM algorithm did reduce the estimation error of the model, and thus improve the accuracy of S-T term alignment, iteratively.

Table 1: Iteratively improved contextual similarity with EM for synonymous term pairs.

S-T Term Pairs	iteration0	iteration1	iteration2	iteration3	iteration4	iteration5
互聯網:網際網路	-5.80539	-3.92528	-2.76368	-1.88505	-1.22829	-0.75455
打印機:印表機	-4.72155	-3.01915	-2.10074	-1.39606	-0.85826	-0.46924
激光:雷射	-4.45118	-3.01697	-2.41179	-1.9353	-1.53044	-1.16807

3.3 Comparison with Word-Based Exact Match Models

To know the improvement in SC-TC term alignment accuracy, the same contextual windows data set used in (Chang and Kung, 2007) is used in order to compare the EM-based results with the previous work in (Chang and Kung, 2007). Chang and Kung (2007) used 4 different criteria in measuring the similarity between two contextual windows. Such similarity scores are kinds of expected counts in proportional to the number of matched contextual words or characters within the words; such highly simplified expected counts are used in turn to estimate the joint probabilities of S-T pairs. In particular, the Lw-Rw model (Word-based matching against left/right contexts) estimates similarity between two contextual windows based on words in the left & right contexts; each match of a contextual word contributes a constant expected count.

The EM algorithm virtually runs in the Lw-Rw mode of operation and uses the same contextual words for estimating the translation probability. Therefore, this criterion is the major criterion for comparison. Furthermore, the results of the 4-th EM iteration are used for comparison since it almost converges there. The results are shown in Table 2. The label "S2T" (S-to-T) means to search TC terms for SC terms, "T2S" means to translate in the T-to-S direction. The two searching methods can be combined using a global optimization strategy suggested in (Chang and Kung, 2007). This strategy is represented with the "S2T+T2S" label. The last column (Δ%) shows the improvement of the EM algorithm over the previous work in (Chang and Kung, 2007), in terms of accuracy rate (Acc).

In comparison with the Lw-Rw model of the previous work, it is clear that the EM model outperforms significantly in all the three different domains. The EM algorithm can achieves 47%~85% accuracy, and is better than the previous work by 2~48%. In particular, the improvement in the transliteration domain (D2) is surprisingly impressive. This can be attributed to the fact that direct pattern matching for transliterated names is not effective since there is not a single way to transliterate a name. These comparisons clearly indicate that the EM re-estimation algorithm greatly boost the SC-TC term alignment performance.

Table 2: Comparison of EM with previous word-based matching using Lw-Rw model.

		Lw-Rw		EM (Lw-Rw)		
domain	Direction	#Correct	%Acc	#Correct	%Acc	△%
D1	S2T	86/136	63.20%	96/136	70.59%	+7.39%
	T2S	73/136	53.70%	99/136	72.79%	+19.09%
	S2T+T2S	79/136	58.10%	102/136	75.00%	+16.90%
D2	S2T	27/82	32.93%	63/82	76.83%	+43.90%
	T2S	33/82	40.24%	66/82	80.49%	+40.24%
	S2T+T2S	31/82	37.80%	70/82	85.37%	+47.56%
D3	S2T	52/128	40.63%	60/128	46.88%	+6.25%
	T2S	58/128	45.31%	61/128	47.66%	+2.34%
	S2T+T2S	52/128	40.63%	64/128	50.00%	+9.38%

4 Concluding Remarks

Searching is a necessary process for utilizing the web resources. It is not only useful for conventional information retrieval and extraction applications but also for utilizing the Web as a huge linguistic corpus. Unfortunately, both searching targets and their contexts might be ambiguous or be expressed in synonymous terms or translation equivalents; such ambiguity or

equivalence must be correctly disambiguated or aligned in order to mine useful resources and miss almost nothing.

A statistical context-based searching and disambiguation model, based on an unsupervised EM algorithm, is proposed in this work. The proposed context-based searching model provides an automatic framework for handling searching problems that utilize contexts for *disambiguation* and flexible *synonymous matching*. The underlying EM algorithm for parameter estimation relieves the need for exact matching against the contexts, and improves searching performance significantly. Various exact or partial match models in previous work are not robust across domains; the current model which is capable of synonymous matching largely overcomes such problems.

The proposed model formulates the searching problem as a machine translation problem in which pieces of contexts are accumulated to enforce the translation probability for a search result to be the translation of the source query. Such a formulation is also likely to provide better syntactic constraints for disambiguation than other models which use the distribution of contextual terms for measuring contextual similarity; the reason is that the latter uses a bag of words without considering their distances and directions with respect to the searching target.

This model is applied to the term alignment problem for searching equivalent TC-SC terms, which cannot be done based on direct pattern match. The comparisons of the EM-based re-estimation with the previous character or word based matching methods for measuring context similarity clearly indicate the superiority of the proposed EM algorithm. In the Lw-Rw model, the EM algorithm achieves 47~85% accuracy; and it significantly outperforms word-based exact matching models by 2~48% in various sub-domains.

References

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra and R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics, 19(2)*:263–311.

Chang, Jing-Shin and Chun-Kai Kung. 2007. A Chinese-to-Chinese Statistical Machine Translation Model for Mining Synonymous Simplified-Traditional Chinese Terms. *Proceedings of Machine Translation Summit XI*, pp.81-88.

Chiang, Tung-Hui, Jing-Shin Chang, Ming-Yu Lin and Keh-Yih Su. 1992. Statistical Models for Word Segmentation and Unknown Word Resolution. *Proceedings of ROCLING-V*, pp.123-146.

Dempster, A. P., N. M. Laird and D. B. Rubin. 1977. Maximum Likelihood from Incomplete Data via the *EM* Algorithm. *Journal of the Royal Statistical Society, 39 (b)*, pp.1-38.

Lee, Lillian. 1999. Measures of Distributional Similarity. *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pp.25-32.

Och, Franz J. and Hermann Ney. 2000. A comparison of alignment models for statistical machine translation. In *Proceedings of COLING-2000*, The 18th International Conference on Computational Linguistics, pp.1086–1090, Saarbrücken, Germany, August.

Sproat, Richard and Thomas Emerson. 2003. The First International Chinese Word Segmentation Bakeoff. (http://www.sighan.org/bakeoff2003/bakeoff_instr.html)