情 報 学 基 礎 63−11
（2001．7．25）

空間分割型CL-LSIによる大規模言語横断情報検索

森 辰則
横浜国立大学大学院 環境情報研究院
E-mail: mori@forest.eis.ynu.ac.jp

本稿では，LSI方式による言語横断情報検索において問題となる，大規模な対訳コーパスの利用方法について考察する．大規模対訳コーパスを用いて単語空間を作成しようとすると，LSIの要である単語頻度行列の特異値分解が記憶装置の制約で難しくなるとともに，語の訳の曖昧性が非常に大きくなるという問題がある．そこで，対訳コーパスを文書の類似度に従って，複数の部分対訳コーパスに分割し，各々の単語空間を作成する手法を提案する．この方法では，検索対象の文書は，最も類似した部分対訳コーパスから構成された単語空間に配置することによって，訳語の曖昧性を減少させる．検索時には，検索質問をそれぞれの単語空間に配置し，文書ベクトルとの類似度計算を行なうことにより検索を行なう．この時に，単語空間毎の未知語に対する重み付けの補正が重要であることを示す．

Large-scaled Cross-Language Information Retrieval based on Segmented CL-LSI

Tatsunori Mori
Graduate School of Environment and Information Sciences,
Yokohama National University

In this paper, we report the utilization of a large-scaled bilingual corpus in Cross-Language Latent Semantic Indexing(CL-LSI). When we construct one monolithic word space with a large-scaled corpus, we encounter problems such as the increase in ambiguity of word translation, the difficulty in singular value decomposition, which is the important process in LSI. In order to cope with the problems, we introduce the method in which the large bilingual corpus is divided into smaller sub-corpora according to the similarity among documents in it, and from each of them one word sub-space is created. By placing each document in the word sub-space, which is made from the sub-corpus most similar to the document, ambiguity of translation is expected to decrease. In the searching process, the query is placed into every word sub-spaces, and similarity between the query and the documents are calculated.

1 はじめに

近年，インターネットの発展などにより外国語文書を電子的に入手する機会が急激に増えており言語の壁を越えた情報検索技術である言語横断検索(CLIR)の要求が高まってきている[菊井00]．

言語横断検索において現在主流の方法は何らかの形で対訳辞書を用いている．そこでは辞書作りの過程で人間が対訳情報を吟味しているので，対訳辞書を用いる手法では翻訳に関する精度が良いと考えられる．その精度は対訳辞書の質や規模のみならず，その使用方法に依存するところが大きい．

一方，対訳コーパス等の言語資源から対訳に関する情報を自動的に抽出し，言語横断検索に利用する研究がある．これらの手法は，対訳辞書を用いずに言語横断検索が可能であるという点で魅力的であり，ある程度の精度で検索ができることが報告されている．その手法としては，対訳辞書を自動生成し，検索質問の翻訳に役立てるといった手法や，ベクトル空間法などにおける文書の中間表現の作成の際に利用する方法などがある．後者の手法として代表的なものが，Cross-Language Latent Semantic Indexing(CL-LSI)である．Carbonellら[CYF+97]によると，中規模コーパス(1134対訳)を用いた場合においては，事例に基づく機械翻訳による検索質問翻訳手法が最も性能が良く，CL-LSI手法とGVSM手法がこれに次ぎ，最も性能の悪かった手法が一般対訳辞書を用いた検索質問翻訳手法であった．

しかし，大規模な文書集合を検索対象とする場合に，CL-LSIなどの手法が，どのような性能になるかは未だ不明である．特に，CL-LSIは，大規模対訳コーパスに対処できないという問題点がある．すなわち，大規模文書集合を検索対象とするためには，必然的に同等の範囲・規模を持つ大規模対訳コーパスを用いて，翻訳に関する情報を抽出しなければならない．一方，CL-LSI方式では単語空間を作成する際に大規模な対訳コーパスを用いると，LSIの要である単語頻度行列の特異値分解が記憶装置の制約で

難しくなる．更には，複数の分野に跨っての頻度累計を行なうため，訳語の曖昧性が非常に大きくなる．

そこで，本稿では，対訳コーパスを文書の類似度に従って，複数の部分対訳コーパスに分割し，各々の単語空間を作成する手法を提案する．この方法では，検索対象の文書は，最も類似した部分対訳コーパスから構成された単語空間に配置することによって，訳語の曖昧性を減少させる．検索時には，検索質問をそれぞれの単語空間に配置し，文書ベクトルとの類似度計算を行なうことにより検索を行なう．この時に，単語空間毎の未知語に対する重み付けの補正が重要であることを示す．

2 Cross-Language Latent Semantic Indexing

Cross-Language Latent Semantic Indexing(CL-LSI) は，検索質問の翻訳を必要としない，完全自動の言語横断検索手法である [DLL96, DLLL97]．これは複数の言語を含む「意味」空間を Latent Semantic Indexing(LSI) により自動的に生成することによりなされる．

2.1 Latent Semantic Indexing

Latent Semantic Indexing の特徴は，語を次元とする典型的なベクトル空間法ではなし得ない，語と語の間の関係を自動的にモデル化し，検索効率を向上させることにある [DDH90]．これは，単語間の直接の対応がほとんど役に立たない言語横断型検索において，非常に重要なものである．LSI では，まず，語-文書頻度行列を作成する (図 1 左側)．この行列の要素 (i, j) は文書 j における語 i の頻度である．行列の各行は，ある単語が文書中にどのように出現したかを表す情報であるから，その単語の出現する文脈を表現すると考えられる．これら文脈から単語間の重要な連想関係を発見するために，線形代数の手法である特異値分解 (singular Value Decomposition) が語-文書頻度行列に適用される．これにより，似通った文脈に登場する語が近くに配置されるように次元を縮退した特徴量空間が形成される．これを LSI 空間と呼ぶ．通常のベクトル空間法では各語は文書ベクトルの各次元に対応するので，各語は互いに直行するベクトルとして表現される．一方，LSI においては，語は文脈を表す語ベクトルによって表現されるので，必ずしも，語ベクトルの間に線形独立性はない．二つの語が似通った文脈で使われている場合には，LSI 空間において類似したベクトルとして表現される．

新しい文書や検索質問文は，それを構成する語のベクトルの重み付き線形和により，LSI 空間に畳み込まれる．検索は，検索質問文と文書との間の類似度を cosine 相関度などにより計算し，順位付けを行なうことで達せられる．

2.2 LSI による言語横断検索

LSI は簡単に言語横断検索に適用できる．CL-LSI では対訳文書集合から単語ベクトル (LSI 空間) を作成する訓練段階と，(一般に対訳ではない) 検索対象文書を LSI 空間に配置し検索を行なう段階に別れる．訓練段階においては，対訳文書対を一つの文書に見たて，その文書に出現する単語の頻度を求め，単語-文書行列を得る．この行列に対して LSI と同様に SVD を適用し，単語ベクトルを得る (図 1 右側)．訳語となる単語同士では，文書対における出現の分布が似ているので，LSI 空間上の近い位置に配置されることが期待される．この性質により，言語を横断することが可能となる．

対訳になっていない検索対象文書は，LSI における新文書の扱いと同様であり，構成単語により LSI 空間に畳み込む．

3 大規模コーパスにおける CL-LSI の問題点

CL-LSI 方式は文書の扱う対象領域がある程度限定されている場合に有効な手法であると考えられる．対象領域が広範囲に亙る場合に，LSI 空間を作成するにあたって，未知語の出現確率が低くなるように対訳文書対を集めるとすると，文書対の数が大きくなる．これは SVD において計算量の問題を生じさせる．SVD は行列操作であるので，行列の次元が高くなれば，それに応じて記憶資源を消費する．よって，語彙数を高くするために非常に大きな対訳コーパスを使おうとすると，空間計算量において破綻する．

例えば，NTCIR1[NTC00] で公開された論文要約の対訳コーパスにおいては，約 18 万対訳 (180802) があり，単語数は複合語などを含めると約 37 万語であった．これを行列の要素数にすると，67×10^9 要素であるから，0 要素が多く含まれていたとしても，中規模程度の計算機の主記憶装置に入れることは事実上不可能である．

そこで我々は，訓練のために用いるコーパスを計算機の資源に併せて分割し，各々の部分コーパスから別々な LSI 空間を生成する方法を提案する．その枠組を図 2 に示す．

コーパスの分割により，まず，SVD が可能となり，更に，各部分コーパス毎に分野がある程度限定されていれば，訳語の曖昧性の減少に役に立つと期待される．しかしながら，訓練コーパスを分割するにあたって，少なくとも，以下に示すような検討事項がある．

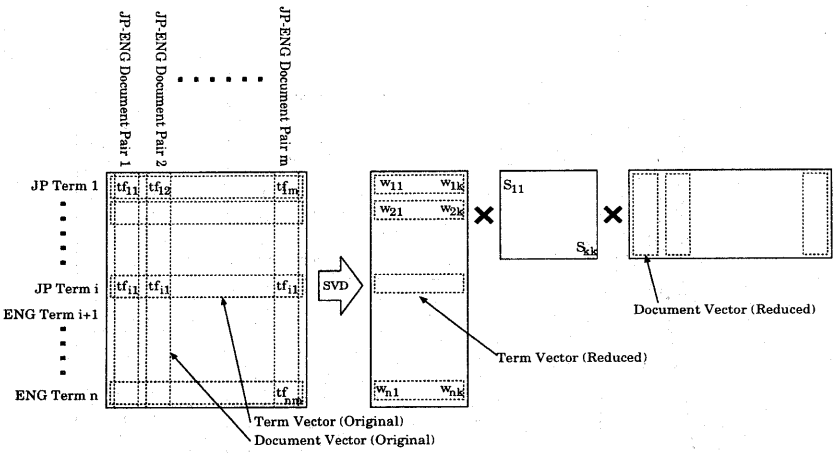

図 1: Cross-Language Latent Semantic Indexing

- どのようにコーパスを部分コーパスに分割するか.
- 得られた複数の LSI 空間に対して，どのように，文書を配置するか.
- 配置された文書をどのように検索するか.

次節では，上記検討事項に注意しつつ，我々の枠組について述べる.

4 大規模対訳コーパス向けCL-LSI

4.1 LSI 空間の分割

部分コーパスにおいて，その中の文書の分野が限定されていると，文脈も自ずと限定され，個々の単語における対訳の多様性も軽減されると考えられる. よって，複数の LSI 空間を構築するにあたっては，類似度に従って対訳文書を複数の部分グループに分割することが有効と考えられる.

これを自動的に行なう手法としては，各種クラスタリングアルゴリズムが知られている. しかし大規模な対訳文書群に対してクラスタリングを行なうには，多大な計算機資源が必要とされる[1]. また，いくつの文書部分集合に分割すればよいかについては，利用可能な計算機資源に依存するので最終的な調整は人間の手によるところが大きい.

そこで，我々は別の手法を検討する. 実際の文書には分野を特定するのに有効情報が付加されていることも多い. 例えば学術論文を考えると，個々の文書に学会名などの分野の名前 (分野名) に関する情報が付与されているのが通常である. この情報を利用することにより，本稿では以下に述べる半自動的なクラスタリングを用いる. すなわち，同じ分野名を持つ文書対を一つのグループと考えグループを併合・分割することにより，適切な大きさのグループを作成する.

1. 対訳文書を分野名によって分類し，分野グループを作成する.
2. 各対訳文書の文書ベクトルを作成する. このベクトルは語を次元とし要素を対応する語の tfidf 値とする.
3. 同一分野グループ内の文書ベクトルの平均を求め，それを分野ベクトルとする.
4. 文書数の多い分野グループを数個，手作業で選択し， 主要分野グループとする.
5. 残りの分野グループの各々について，最も類似度の高い主要分野グループに併合する. 類似度は，分野ベクトルの間の方向余弦により求める.
6. 文書数が上限よりも大きくなってしまった分野グループは，上限を満足するように分割する. 文書数の上限は計算機資源により決める.
7. 各分野グループの分野ベクトルを更新する.

4.2 検索文書の配置

本手法においては LSI 空間は分野グループによって異なるので，どの LSI 空間に文書を配置するかによって，異なる文書ベクトルが作成される. 複数の

[1] BIRCH アルゴリズム [ZRL96] に代表されるように，近年，大規模データのクラスタリング手法が提案されてきているので，この傾向は緩和されつつある.

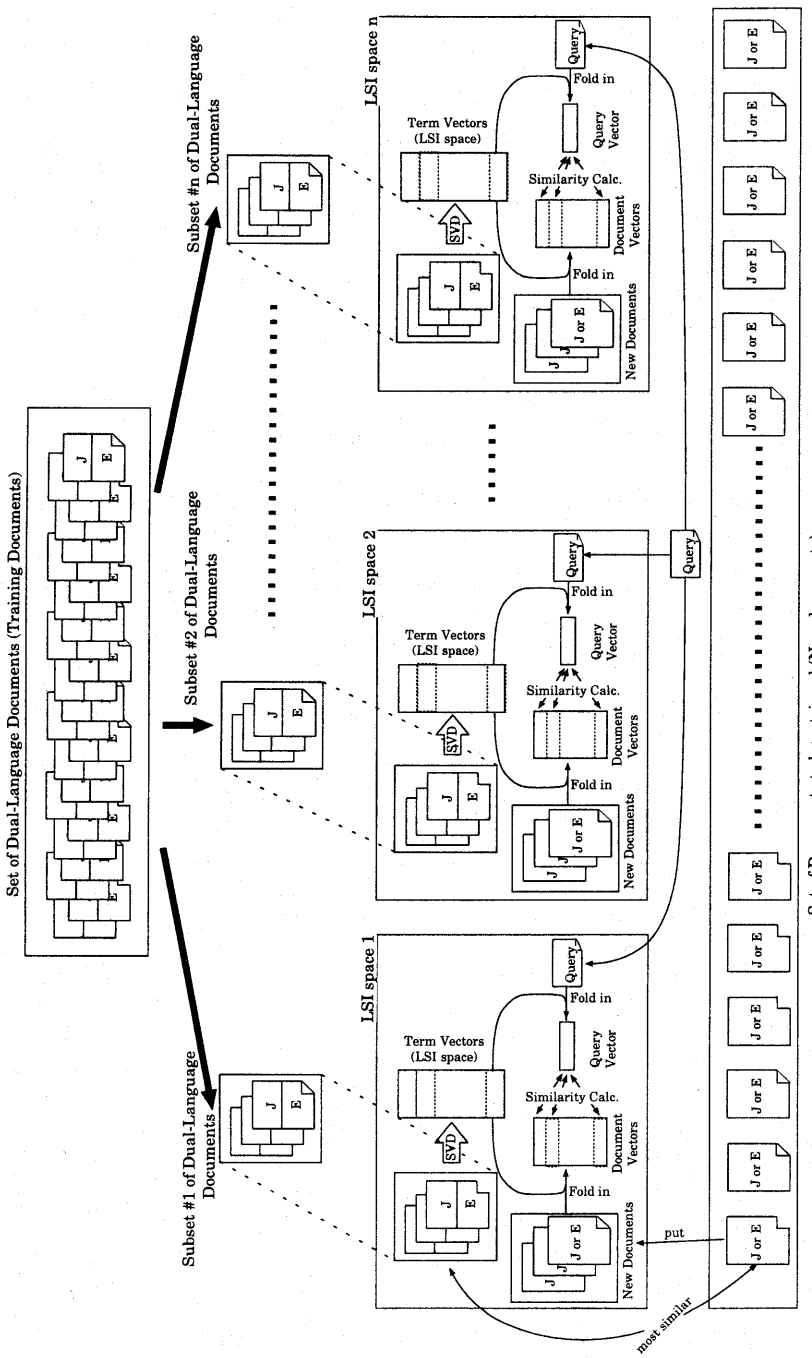

図 2: 空間分割型 CL-LSI の枠組

LSI空間がある状況において，文書ベクトルを作成する方法には主に，すべてのLSI空間に配置する方法と，一つのLSI空間を選択し，そこに配置する方法が考えられる．前者の方がすべての翻訳情報を利用するので，検索の精度が良いと考えられるが，LSI空間の数だけ個別の文書ベクトルが必要であり，記憶装置もその分占有してしまう．これは特に大規模文書データベースを作成する時に問題となるので，本稿では選択した一つのLSI空間のみに配置する．

この方式では，文書ベクトルを配置するLSI空間を適切に選択する方法を考えなければならない．ある文書を配置先は，訳語選択の分野依存性から，同一の分野の対訳から作成されたLSI空間であることが望ましい．よって，各文書を最も類似する分野ベクトルを持つ分野のLSI空間に配置する．

文書をLSI空間に畳み込む方法としては次の式を用いる[2]．

$$\mathbf{D} = \sum_{T_i \in D} tf(T_i, D) idf(T_i) \mathbf{T}_i$$

D	:	文書Dのベクトル
$tf(T_i, D)$	:	文書D中の語T_iの頻度
$idf(T_i)$	:	語T_iのidf値 $\log \frac{N}{df(T_i)} + 1$. $df(T_i)$はT_iの文書頻度. Nは総文書数
$\mathbf{T}_i$	:	語T_iのベクトル

4.3 複数LSI空間での文書検索

CL-LSI方式では検索質問も他の文書と同様にLSI空間上のベクトルとして表現される．検索質問に対する各文書の順位づけは，検索質問ベクトルと間の類似度に基づき行なわれる．しかし，我々の方法ではLSI空間が複数あるので，以下の手順により文書検索を行なう．

1. 検索質問をすべての文書と比較するために，各LSI空間に対して，検索質問ベクトルを一つずつ作成する．
2. 各LSI空間毎に，検索質問ベクトルとすべての文書ベクトルの間の類似度を計算する．
3. その類似度を，複数のLSI空間に亙って，降順に整列することにより文書に対する順位づけを行なう．

5 対訳コーパスの分割による未知語の問題

CL-LSI方式では対訳コーパス中に現れない単語については，その対訳情報が得られないので，文書ベクトルを作成するときに無視される．よって，検索対象文書中には現れるが，対訳コーパスに現れない語が検索質問中に含まれるときには，検索精度が低下する．これは原理上不可避である．

一方，我々の方式には，これとは異なる種類の未知語の問題がある．それは，対訳コーパスを分割することにより生じたものである．コーパスを分野毎に分割した場合，関連する部分コーパスのみに現れ，他の部分コーパスに現れないような語が存在しうる．このため，LSI空間毎に未知語が異なる可能性があるので，文書検索において期待どおりの結果が得られないことがある．

例として，T_a，T_b，T_cという3つの語からなる検索質問$Q(T_a, T_b, T_c)$でLSI空間TS_1とTS_2中の文書を検索する場合を考える．LSI空間TS_1には，語T_aが存在し，T_b，T_cが未知語となっているとする．そこには，T_aのみが含まれる文書$D_1(T_a)$が配置されているとする．また，LSI空間TS_2には，語T_a，T_b，T_cのすべてが存在するとし，ここに，T_a，T_bが含まれる文書$D_2(T_a, T_b)$を配置するとする．

ここでは当然，$D_1(T_a)$よりも$D_2(T_a, T_b)$の順位を高くしたく，QとD_2の間の類似度の方がD_1の場合よりも大きくなることを期待する．しかし，上述の状況においては，全く逆で，D_1の類似度のほうがD_2よりも大きくなってしまう．これはTS_1での類似度計算は，T_aのみについて行なわれるので，検索質問があたかもT_aであると見なされるためである．一方，TS_2での類似度計算は，T_a，T_b，T_cについて行なわれるので，検索質問と文書D_2の間の類似度はD_2がT_cを持たない分，低くなる．

この例が示すとおり，我々の望む類似度計算を行なうためには，検索質問中の未知語を単に無視するのではまずく，類似度を低下させる要因として適切に扱わなければならない．一方，未知語に関しては，元々その情報がないので個別の対応ができない．そこで，我々は，任意の未知語に対応する新しい次元を一つ，各LSI空間に導入する方法を提案する．この方法は，次に述べる手順でLSI空間を拡張し，未知語を既存のどの単語ベクトルとも直行するベクトルとして扱うことにより，類似度に対する補正を行なうものである．

あるLSI空間がn次元空間で表現されているとする．すると，単語は$(w_1, ..., w_n)$なるベクトルになる．この空間に対して，新たな次元を導入し，空間を(n+1)次元とする．このとき，既存の単語については，$(w_1, ..., w_n, 0)$とし，一方，検索質問に現れるすべての未知語を，$(0, ..., 0, 1)$とする．この時，文書ベクトルにおいて，(n+1)次元目の成分が常に0であることを考慮すれば，h補正の前後で以下の各式が成り立つ．

$$\begin{aligned} \mathbf{D}' \cdot \mathbf{Q}' &= \mathbf{D} \cdot \mathbf{Q} \\ |\mathbf{D}'| &= |\mathbf{D}| \end{aligned}$$

[2] idf値を導入している点で，元のLSI方式とは異なる．

$$
\begin{aligned}
|\mathbf{Q}'| &= |\mathbf{Q}+\mathbf{Q_u}| \\
&= \sqrt{|\mathbf{Q}|^2+(\sum_{T_i\in Q_u} tf(T_i,Q_u)idf(T_i))^2}
\end{aligned}
$$

ここで,

$\mathbf{D}$, $\mathbf{D}'$:補正前後の文書ベクトル
$\mathbf{Q}$, $\mathbf{Q}'$:補正前後の検索質問ベクトル
Q_u: 検索質問中の未知語のリスト
$\mathbf{Q_u}$: Q_u に対応する部分の検索質問ベクトル

である．よって，補正後の文書ベクトル D' と検索質問ベクトル Q' の間の類似度 $sim(D',Q')$ は，次式となる．

$$
\begin{aligned}
sim(\mathbf{D}',\mathbf{Q}') &= \frac{\mathbf{D}'\cdot\mathbf{Q}'}{|\mathbf{D}'||\mathbf{Q}'|} \\
&= \frac{\mathbf{D}\cdot\mathbf{Q}}{|\mathbf{D}|\sqrt{|\mathbf{Q}|^2+(\sum_{T_i\in Q_u} tf(T_i,Q_u)idf(T_i))^2}}
\end{aligned}
$$

この式をみると，本補正により，新たに構築された空間での類似度は，検索質問に文書に含まれている未知語の分だけ低く見積もられることがわかる．

6 評価実験

6.1 索引語の認定

本稿での実験においては，以下の手法により索引語を選択している．

索引語には単単語ならびに複合語を用いた．日本語文書については，形態素解析器 JUMAN 3.61 により，単語切り出しならびに品詞付与を行なった．品詞情報により，名詞，形容詞，動詞，連体詞，副詞，アルファベット，カタカナを取り出した．英語については，Frakes の方法 [Fra92] でステミングを行なうとともに，Fox[Fox92] に基づきストップワードリストによる不用語の削除を行なった．

複合語の認定には，日英両言語で一貫して扱える手法として，C value に基づく方法を用い，次の二段階で行なった [FS96]．まず，対訳コーパスより，語を単位とするサフィックスアレイを作成し，単語列の出現頻度を求めた．そしてある閾値 TH_f 以上の出現回数をもつ単語列を複合語の候補とした．次に各複合語候補に対して，C value を計算し，その値がある閾値 TH_c 以上のものを複合語として認定した．今回の実験では，プログラムの制約上，NTCIR1 の対訳コーパスを 11 に分割し，各部分集合において，$TH_f=5$,$TH_c=5$ の条件の下で，上記手続きを行なった．なお，複合語の構成要素も索引づけに用いている．

6.2 LSI 空間の分割

LSI 空間を複数に分割して作成し検索を行なう我々の方式では，対訳コーパスの情報を一度に参照していないという点で，単一 LSI 空間を用いた方式より精度が劣る可能性がある．一方で，分野毎に LSI 部分空間を作成すれば，分野に応じて対訳情報が得られる可能性があるので，精度が向上する可能性もある．そこで，この点を確認するために，手元の計算機環境で単一空間を構成できる範囲で，メイト検索[3]による評価実験を行なった．

まず NTCIR1 の言語横断タスクから得られた学会情報付の対訳技術文書 (要約) 6000 対を訓練コーパスとして用いた．その訓練コーパスを学会情報を基に 3 つに分割し，部分 LSI 空間を作成し提案手法により検索した場合を，すべての訓練コーパスにより一つの LSI 空間を作成・検索を行なった場合と比較した．検索対象として訓練コーパスとは別の対訳文書 (3000 対) を用いた．結果を表 1 に示す．

表 1: 分割 LSI 空間の精度評価

LSI 空間	1 位 (%)	3 位 (%)
全体	58.2	75.7
分割	47.8	63.9
分割 補正後	59.4	78.2

6.3 NTICR2 における実験

大規模な文書集合に対する実験として，NTICR2 における言語横断タスクにおいて評価を行なった [NTC00]．検索規模は非常に大きく，提案手法が大規模な検索タスクでどの程度の精度となるのかを評価できる．訓練コーパスとしては NTCIR1 で使用された日英技術文書要約約 38 万件を使用することが可能で，検索対象は日英技術文書要約約 70 万件であった．

本実験では上記コーパスから日英の対訳対が得られた約 18 万文書対を訓練用に使用した．これは，57 学会の論文要旨の対訳対からなるもので，次の手順で部分コーパスに分割した．まず，最初に要旨対数の最も多い 6 つの学会を選択し主要学会グループとした．ついで，残りの学会をそれぞれのクラスタに配置した．計算機資源を考慮して，そのうちの 4 クラスタを 2 つに分割し，最終的に 10 の文書集合を得た．各集合の大きさは約 14000 から約 26000 文書対であった．また，単語の種類数は約 78000 から約 115000 であり，全体では約 380000 語であった．

[3]検索対象文書集合から文書を一つ選択し，その対訳を検索質問とする．このとき，元の文書が何位で検索されるかをみることにより，検索精度を検証する手法．

検索トピックとしては日英各々49件あり，我々の実験は，DESCRIPTION フィールド (1 文程度) 単体を検索質問とした場合と，DESCRIPTION と NARRATIVE(要約文書程度) を合わせたものを用いた場合で行なった．結果を表 2 に示す．'J-E' は日本語を検索質問とし，英語文献を検索対象とする場合であり，'E-J' はその逆である．'Desc' は DESCRIPTION フィールドを検索質問とした場合であり，'Desc-Nar' は NARRATIVE フィールドを検索質問とした場合である．

表 2: 全ての検索質問に対する平均適合率，R 適合率

	Average precision	R-Precision
J-E-Desc	0.0533	0.0635
補正後	0.0666	0.0786
補正による改善	24.9 %	23.8 %
J-E-Desc-Nar	0.0868	0.1031
補正後	0.0940	0.1096
補正による改善	8.3 %	6.3 %
E-J-Desc	0.0512	0.0705
補正後	0.0610	0.0839
補正による改善	19.1 %	19.2 %
E-J-Desc-Nar	0.0609	0.0876
補正後	0.0736	0.1018
補正による改善	20.8 %	16.2 %

本方式ではどの訓練コーパス上にも出現しない語については，単語ベクトルが存在しないのでその語が検索質問に出現した場合に検索精度が悪くなる．そこで未知語のない検索質問においてどの程度の精度が見込まれるかを別途評価した．結果を表 3 に示す．

7 考察

一つめの実験によれば，LSI 空間を分割した場合，分割しない場合に比べて，大幅な精度低下が見られる．しかしながら，更に未知語の効果を補正すると，同等もしくは若干の精度の向上が見られる程度まで性能が改善することがわかる．よって，中規模実験ながら，空間分割型 LSI では，未知語の補正が必要であること，また，単一空間による LSI よりも性能が低くならないことが確認できた．いずれも，期待したとおりである．

次に二つめの実験について考える．絶対的な性能評価の観点からすると，やはり人手で構築した対訳辞書に基づく手法に比べ，検索精度がかなり低いといわざるをえない．NTCIR2 における最もよいシステムの平均適合率が 30%を越えているのに対し，我々の手法では約 10%である．しかし，ある

表 3: 未知語のない検索質問に対する平均適合率，R 適合率

	検索質問数	Average precision	R precision
J-E-Desc	43	0.0600	0.0704
補正後	43	0.0743	0.0870
補正による改善		23.8 %	23.6 %
J-E-Desc-Nar	31	0.1032	0.1206
補正後	31	0.1094	0.1307
補正による改善		6.0 %	8.4 %
E-J-Desc	43	0.0579	0.0782
補正後	43	0.0692	0.0942
補正による改善		19.5 %	20.4 %
E-J-Desc-Nar	39	0.0738	0.1025
補正後	39	0.0872	0.1187
補正による改善		18.1%	15.8 %

程度の規模の対訳コーパスがあれば，大規模な言語横断検索も可能であるということが確認された．また，我々の導入した補正手法の効果は大規模コーパスを対象にした場合でも確認される．特にそれは DESCRIPTION フィールドだけを用いた場合のほうが，DESCRIPTION ならびに NARRATIVE フィールドを用いた場合よりも顕著である．これは，短い検索要求に未知語が現れた時には，有効な語の影響が相対的に大きくなってしまうためである．

一方，コーパスに全く現れない完全なる未知語の影響についてみると，コーパスに現れる単語のみからなる検索質問に限定した場合に，精度が向上している．やはり，完全な未知語については，LSI に基づく方式で不可避の問題として依然として残ることがわかる．

さらに，我々と同様にコーパス情報のみを用いた手法により NTCIR2 に参加したシステムと比較してみる．Jiang ら [JL01] は，Approximate Dimension Equalization という手法を導入している．この手法は，より少ない特異ベクトルの計算により，LSI の持つ効果を達成するものである．NTCIR2 の評価実験においては，NTCIR1 の対訳文書を学習用に用いた場合，J-E ならびに E-J の平均適合率が，それぞれ，0.0724，0.0829 であることが報告されている．両者の平均は 0.0777 である．この実験においてどのフィールドを検索質問に使用しているかは不明であるが，我々の結果が，J-E,E-J の平均で 0.638(DESCRIPTION フィールドのみ)，0.838(DESCRIPTION+NARRATIVE) であるから，ほぼ同等の性能と考えられる．

対訳コーパスを用いる両手法において，ほぼ一致したさほど高くない精度しか得られなかったことから，NTCIR1 コーパスから得た対訳コーパスが

NTCIR2で新たに加わった検索対象文書に適合していなかったことが考えられる．そのような状況に対応できないという意味において，これは，対訳コーパスのみによる手法の限界を示しているであろう．

8 おわりに

本稿では，既存の対訳コーパスのみを翻訳の情報として用いる情報検索手法として，CL-LSI手法に注目した．我々は，これを大規模対訳コーパスに適用するために，訓練用の対訳コーパスを分割し，複数のLSI空間を併用する方法を提案した．また，LSI空間毎に異なる未知語に起因する検索精度低下について検討し，対処方法を提案，その効果をメイト検索により確認した．さらに，NTCIR2における評価実験により，大規模文書集合を対象とした検索においても，同手法が適用できることを示し，他の対訳コーパス方式と同等の性能が得られることが確認された．ただし，その性能は対訳コーパスに基づかない他の手法に比べると低い精度に留まっている．

一方，今回の実験ではLSI空間の分割による精度向上がメイト検索ではない実際の検索の場面において，精度向上に役立っているかは不明なままなので別途実験を行ない確認する必要がある．またGVSMなどの他の類似方法で大規模コーパスを用いた場合との比較をする必要がある．

謝辞

本研究を進めるにあたって，東芝(株)の國分智晴氏に多大なる御協力を頂きました．ここに感謝いたします．また，国立情報学研究所主催のNTCIRを企画・運営し，評価用データを作成していただいた皆様にも感謝致します．

参考文献

[CYF+97] Jaime G. Carbonell, Yiming Yang, Robert E. Frederking, Ralf D. Brown, Yibing Geng, and Danny Lee. Translingual information retrieval: A comparative evaluation. In *Proceedings of International Joint Conference on Artificial Intelligence '97 IJCAI '97*, 1997.

[DDH90] Scott Deerwester, Susan T. Dumais, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the Society for Information Science*, Vol. 41, No. 6, pp. 391–407, 1990.

[DLL96] Susan T. Dumais, Thomas K. Landauer, and Michael L. Littman. Automatic cross-linguistic information retrieval using latent semantic indexing. In *SIGIR '96 – Workshop on Cross-Linguistic Information Retrieval*, pp. 16–23, 1996.

[DLLL97] Susan T. Dumais, Todd A. Letsche, Michael L. Littman, and Thomas K. Landauer. Automatic cross-language retrieval using latent semantic indexing. In *AAAI Spring Symposium on Cross-Language Text and Speech Retrieval*, March 1997.

[Fox92] Christopher Fox. Lexical analysis and stoplists. In William B. Frakes and Ricardo Baeza-Yates, editors, *Information Retrieval – Data Structure & Algorithms*, chapter 7, pp. 102–130. Prentice Hall PTR, 1992.

[Fra92] William B. Frakes. Stemming algorithms. In William B. Frakes and Ricardo Baeza-Yates, editors, *Information Retrieval – Data Structure & Algorithms*, chapter 8, pp. 131–160. Prentice Hall PTR, 1992.

[FS96] K. Frantzi and Ananiadou S. Extracting nested collocations. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING 96)*, pp. 41–46, August 1996.

[JL01] Fan Jiang and Michael L. Littman. Approximate dimension reduction at ntcir. In *Proceedings of NTCIR Workshop 2 Meeting*, pp. 5–179–5–74, 3 2001.

[NTC00] NTCIR Project. NTCIR (NII-NACSIS test collection for IR systems) project web page. http://research.nii.ac.jp/ ntcadm/index-en.html, 2000.

[ZRL96] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: An efficient data clustering method for very large databases. In H. V. Jagadish and Inderpal Singh Mumick, editors, *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, pp. 103–114. ACM Press, 1996.

[菊井 00] 菊井玄一郎. 言語の壁を越えて文書を検索する—クロスランゲージ情報検索—. 人工知能学会誌, Vol. 15, No. 4, pp. 550–558, 7月 2000.