Vol.2009-NL-191 No.17
Vol.2009-SLP-76 No.17
2009/5/22

大規模オープンソース日英対訳コーパスの構築

石坂達也[†1]　内山将夫[†2]
隅田英一郎[†2]　山本和英[†1]

Web 上の翻訳文書と原文書を収集し，文の対応付けを行うことで日英対訳コーパスを構築した．対訳コーパスは主にオープンソースソフトウェアのマニュアルの文で構成されており，対訳文の総数は約 50 万文となった．オープンソースな日英対訳コーパスとしては最大級である．この日英対訳コーパスを学習データに使用し，翻訳実験を行った．その結果，BLEU 値は最大で 44.36 となった．この日英対訳コーパスは公開する予定である．

Building A Large Scale Japanese-English Open Source Parallel Corpus

Tatsuya Ishisaka ,[†1] Masao Utiyama ,[†2]
Eiichiro Sumita [†2] and Kazuhide Yamamoto[†1]

We built a Japanese-English parallel corpus which collected open source manual in the Web. The parallel corpus is constructed mainly sentences of open source software manuals. The corpus contains about 500,000 sentence pairs that were aligned automatically. It's one of the largest open source Japanese-English parallel corpus. We conducted machine translation (MT) experiments. Maximum BLEU score was 44.36. We will publish the parallel corpus.

†1 長岡技術科学大学　電気系
Department of Electrical Engineering, Nagaoka University of Technology
†2 情報通信研究機構　MASTAR プロジェクト
MASTAR Project, National Institute of Information and Communications Technology

1. はじめに

対訳コーパスは自然言語処理の分野において需要が高い．例えば機械翻訳では学習データとして対訳コーパスを利用している．また，テキストマイニングの分野では多言語で記述されたテキストから特定の情報を抽出する場合に対訳コーパスを利用している[1)]．そして，自然言語処理の分野に限らず教育の場でも語学学習者の支援として使用できるため対訳コーパスは有用である．

しかし，大規模な日英対訳コーパスは少なく，文書内容の分野も限られている．現在公開されている日英対訳コーパスには内山ら[2)] が新聞記事から作成した日英対訳コーパスがあり，総対訳文数は約 18 万文である．NTCIR-7[⋆1]は特許文で構成された 180 万文の日英対訳文を公開予定としている．このような背景の中で，我々は公開されている対訳文と分野が異なる大規模な日英対訳コーパスの構築を試みた．

日英対訳コーパスを作成するには大量の日本語文と英語文が必要であり，対訳文を人手で作成するには膨大な時間と労力が必要である．Web にはボランティア翻訳者によって翻訳された文書が大量に存在する．この翻訳文書と原文書を収集し，文の対応付けを行うことで対訳コーパスは作成できる．本稿では翻訳文書を日本語，原文書を英語とする．しかし，翻訳文書や原文書には著作権があるため無断での使用，編集，配布は原則的に禁じられている．そのため，Web で公開されている対訳文書の全てを対象にはできない．Web で公開されている対訳文書の中には，自由に入手でき，再配布を可能としているオープンライセンスな文書があり，オープンライセンスな文書は様々な条件の下で公開されている．ライセンスの条件に従うことで文書の編集と配布が可能となる場合がある．

オープンライセンスな文書にはソフトウェアのマニュアルなどがあり，マニュアル翻訳はプロジェクトとして活動をしている場合が多い．翻訳プロジェクトでは翻訳者同士で翻訳物を添削し合っている．また，難解な文を翻訳する時はメーリングリストに投稿し議論している．プロジェクトの翻訳文書の翻訳精度は信頼できると言える．我々はオープンソースソフトウェアのマニュアルの文書を収集することで公開可能な日英対訳コーパスを構築する．

マニュアルの対訳コーパスを構築することにより，以下のことが可能になる．

(1) 対訳コーパスを利用して，対訳検索システムを作成することにより，翻訳者を支援することができる．

⋆1 http://research.nii.ac.jp/ntcir/index-ja.html

Vol.2009-NL-191 No.17
Vol.2009-SLP-76 No.17
2009/5/22

(2) 対訳コーパスを利用して，機械翻訳システムの研究をし，その研究成果をマニュアルの翻訳に利用することができる．

ところが，現状では，マニュアルの対訳コーパスで一般に利用可能なものは日英に関しては大規模なものは存在しない．そこで，我々は一般に利用可能な日英対訳コーパスを構築する．

2. 関連研究

Jörg ら[3] は多国語の対訳コーパスを構築し，公開している．言語資源は OpenOffice.org documentation[⋆1]，KDE manuals including KDE system messages[⋆2]，PHP manuals [⋆3]などである．我々は Jörg らが収集していない対訳文書も収集した．

内山ら[2] は読売新聞と The Daily Yomiuri の文の対応付けを高精度で行う手法を提案すると共に約 18 万文の日英対訳コーパスを構築した．

Koehn[4] は欧州言語の対訳文書を収集し，110 言語対の対訳コーパスを構築した．しかし，欧州言語のみで構成されているため日本語は含まれていない．欧州や他の国では国会議事録などを多言語で記述している．そのため，比較的対訳文書を入手でき，大規模な対訳コーパスを構築できる．しかし日本には対訳文書が少なく，対訳コーパスを構築することは困難である．よって，日英対訳コーパスの構築することには価値がある．

3. マニュアルとライセンス

オープンソースソフトウェアのマニュアルは Web 上に大量に存在し，マニュアルの多くはライセンスの下で公開されている．我々は構築した対訳コーパスを公開する．そのため，編集物の配布が許可されているライセンスの文書を対象にして収集しなければならない．

3.1 ライセンスの例

編集物の配布が許可されているライセンスを紹介する．

3.1.1 MIT ライセンス

MIT ライセンス[⋆4]は誰でも無償で無制限に扱って良いとされている．しかし，著作権表示を複製物の全てか重要な部分に記載しなければならない．また，作者や著作者は何らの責任も負わないとする免責事項も記載させている．以下に原文を示す．

MIT ライセンス原文

Copyright (c) <year> <copyright holderu >
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

3.1.2 Creative Commons

Creative Commons[⋆5]にはいくつかの種類がある．その中でも本稿では “表示-継承 3.0” を紹介する．

“表示-継承 3.0” では原著作者の著作権表示を記載すること，二次的著作物も同じまたは類似したライセンスの下で公開することを条件として配布，展示，実演，二次的著作物の作成を許可している．一般向けに公開されている Web ページに記述されている条件を以下に示す．

1. Attribution.

⋆1 http://www.openoffice.org
⋆2 http://i18n,kde.org
⋆3 http://www.php.net/download-docs.php
⋆4 http://www.opensource.org/licenses/mit-license.php
⋆5 http://creativecommons.org/licenses/by-sa/3.0/deed.en

Vol.2009-NL-191 No.17
Vol.2009-SLP-76 No.17
2009/5/22

You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

2. Share Alike.
If you alter, transform, or build upon this work, you may distribute the resulting work only under the same, similar or a compatible license.

3.1.3 Linux Japanese FAQ Project

Linux Japanese FAQ Project （JF）[⋆1]は Linux に関する FAQ や解説文書などを翻訳するプロジェクトである．

JF は，配布文書全体に適用されるようなライセンスはないようである．しかしホームページに翻訳文書を扱う際の注意点が明記されている．注意点は次のようになっている．

JF の著作権に関するコメント

1. JF Project のファイルは自由に配布することができますが, 出来る限り最新のものを配布し, その際に再配布の制限を付け加えてはいけません.
2. 著作者および翻訳者に断りなく書籍・雑誌などに掲載してはいけません. 商業目的でない引用などは問題ありません.
3. 誤りを発見した場合はできる限り著作者または翻訳者まで連絡してください．また文書配布時・引用時に, 連絡先がわかるように配慮してください.
4. JF Project としても調査を行いましたところ, 全ての文書が非商用での再配布を許可していることがわかりました.

JF については「2 次的著作物」については明記していないため，JF メーリングリスト等に問い合わせる予定である．

4. マニュアル文書の特徴

マニュアル文書には以下の特徴があり，全自動化は難しく，人手で整備する必要がある．

- 翻訳文書に英語が混在する
  コマンドの説明をする場合，コマンド名が文中に出現する．コマンド名は翻訳されず英語のまま残る．また，プログラム言語のマニュアルにはソースコードが記述されており，翻訳する必要がない部分が多い．
- 原文書と翻訳文書の書式が異なる
  多くの場合は HTML タグを原文と同じように付加して，原文サイトとほぼ同じ表記になるようにしている．しかし，いくつかのマニュアル文書は原文とは異なる表記方法となっている．
- 文書が更新される
  一般的にソフトウェアは頻繁にアップデートされ，新しい機能が追加される．その度にマニュアルも更新され，文が追加されている．そのため，原文書の最新版と翻訳文書の最新版ではバージョンが異なる場合がある．

5. 対訳コーパス構築までの流れ

以下の手順で対訳コーパスを構築する．

(1) 対訳文書の収集
(2) 文書の整形
(3) 文の対応付け

5.1 対訳文書の収集

マニュアルの翻訳文書と原文書を収集するためには，ソフトウェアの Web サイトを見つけなければならない．そこで，検索エンジンを使用し，人手で Web サイトを見つけ出し，オープンソースソフトウェアのマニュアルを収集した．収集したマニュアル文書名と URL を**表 1** に示す．JM は “JM project” と呼ばれる Linux の man ページの翻訳を行っているプロジェクトである．RFC は “Request for Comments” の略で技術仕様書のような文書である．RFC 文書はソフトウェアマニュアルではないが，文書内容がマニュアルに近いため，RFC 文書も収集した．その他はソフトウェアマニュアルである．

5.2 収集した文書の整形

収集した文書には HTML タグが含まれていたり，文の途中に改行がある．収集した文書を整形せずに文の対応付けを行うと，対応付けの精度が低下する．よって，収集した英語文書と日本語文書をそれぞれ 1 行 1 文になるように整形する．

HTML は正規表現で簡単に削除できる．タグの途中に改行が入っている場合も考慮し，タグを削除した．

文の途中にある改行については 2 つの規則を与えて，文の途中かどうかの判定をした．マニュアル文書には章の題目と本文の間に空行が存在する．空行から空行までの行を 1 区分として，1 区分中に含まれる改行は全て削除する．そして，以下の条件により文の途中かど

⋆1 http://www.linux.or.jp/JF/

Vol.2009-NL-191 No.17
Vol.2009-SLP-76 No.17
2009/5/22

表 1 収集したマニュアル文書一覧
Table 1 Compendium of collected manuals

	日本語	英語
FreeBSD	http://www.freebsd.org/ja/	http://www.freebsd.org/
Gentoo Linux	http://www.gentoo.org/doc/ja/index.xml	http://www.gentoo.org/doc/en/index.xml
JF	http://www.linux.or.jp/JF/	http://www.kernel.org/pub/linux/kernel/ http://tldp.org/
JM	http://www.linux.or.jp/JM/	http://www.sfr-fresh.com/ http://www.kernel.org/ http://ftp.gnu.org/gnu/
Net Beans	http://ja.netbeans.org/index.html	http://www.netbeans.org/index.html
PEAR	http://pear.php.net/index.php	http://pear.php.net/index.php
PHP	http://www.php.net/download-docs.php	http://www.php.net/download-docs.php
Postgres	http://www.postgresql.jp/	http://www.postgresql.org/
Python	http://www.python.jp/doc/	http://docs.python.org/download.html
RFC	多くのページから収集	
XFree86	http://xjman.dsl.gr.jp/download.html	http://www.xfree86.org/

うか判定する．

(1) 1 区分の中に句点 , . , ! , ?があるならそれぞれで改行して出力

(2) (1) の記号がないなら題目として処理するために未編集で出力

6. 文の対応付け

英語文と日本語文の対応付けには内山ら[5] の手法を使用する．この手法は高精度で文の対応ができる．本稿では文の対応付けの手法を大まかに説明する．

日本語文書を J，英語文書を E とする．DP マッチングを用いて（J_1,E_1）,（J_2,E_2），...,（J_m,E_m），ように文の対応付けを行う．J_i は J の含まれる文，E_i は E に含まれる文である．日本語文と英語文は 1 対 1 の対応になるとは限らないため，文間対応を 1 対 n，n 対 1（$0 \leq n \leq 5$）まで許した．

6.1 類似度計算

日本語文と英語文に対して形態素解析を行い内容語を抽出する．日本語の内容語を英語へ翻訳し，英語文の内容語との一致数を計る．英語の場合も同様に日本語へ翻訳し，日本語文の内容語との一致数を計る．この 2 言語の一致数の和（SIM(J_i,E_i)）を文の類似度とし，式（1）で文書の類似度を算出する．

$$\mathrm{AVSIM}(J,E) = \frac{\sum_{i=1}^{m} \mathrm{SIM}(J_i,E_i)}{m} \quad (1)$$

AVSIM(J,E) の値が高いほど J と E が類似しているということとなる．また，J と E に含まれる文の数の比率を（R(J,E)）として, 以下の式で求める．

$$\mathrm{R(J,E)} = \min\left(\frac{|J|}{|E|}, \frac{|E|}{|J|}\right) \quad (2)$$

$|J|$ は J に含まれる文の数で, $|E|$ は E に含まれる文の数である．最後に，Score(J_i,E_i) で J_i と E_i の類似度が算出される．

$$\mathrm{Score}(J_i,E_i) = \mathrm{SIM}(J_i,E_i) \times \mathrm{AVSIM}(J,E) \times \mathrm{R}(J,E) \quad (3)$$

Score(J_i,E_i) の値が高い場合は以下のような傾向がある．

- 日本語文 J_i と英語文 E_i は類似している．
- 日本語文書 J と英語文書 E は類似している．
- 日本語文書 J と英語文書 E の文の量が近い．

6.2 対応付け結果

マニュアルから作成した対訳文の例を**表 2** に示す．

各マニュアルごとの対訳文数を**表 3** に示す．合計で約 50 万文となった．この文の数は文間対応が 1 対 1，1 対 2，2 対 1 の対訳文で構成される．1 対 2 と 2 対 1 を含むには理由がある．まず，文の途中の改行を削除しきれず，本来 1 文の文が 2 文となることがあったためである．また，英語から日本語に翻訳する際に 1 文をあえて 2 文に分割して翻訳する場合があったためである．そして，逆に 2 文を 1 文に翻訳する場合もあった．

表 2 対訳文の例
Table 2 Example of parallel sentences

日本語文	英語文
この節では，エラーの処理方法について説明します.	This section describes how errors are handled.
新しいファイルが XML エディタで開きます.	The new file opens in the XML editor.
画像のヒストグラムを ImagickPixel オブジェクトの配列で返します.	Returns the image histogram as an array of ImagickPixel objects.
それがローカルサービスであるかリモートサービスであるかにかかわらず，コール方法は同じようになることに注意しましょう.	Note that the way the call is made looks the same regardless of whether the call is to a local service or a remote one.
メッセージの HTTP プロトコルバージョンを設定します.	Set the HTTP Protocol version of the Message.

Vol.2009-NL-191 No.17
Vol.2009-SLP-76 No.17
2009/5/22

表 3 各マニュアルの対訳文数
Table 3 Number of aligned sentences

	総文数	英語	日本語
		総単語数 (1 文あたりの平均)	総単語数 (1 文あたりの平均)
FreeBSD	10528	156749(14.9)	245780(23.34)
Gentoo Linux	11117	1488461(13.39)	224324(20.17)
JF	122072	1867792(15.30)	2854297(23.38)
JM	41573	483098(11.62)	731045(17.58)
Net Beans	32774	450849(13.76)	682229(20.82)
PEAR	23333	294233(12.61)	446863(19.15)
PHP	67023	639857(9.55)	977281(14.58)
Postgres	22843	396570(17.36)	627994(27.49)
Python	26215	297830(11.36)	499860(19.07)
RFC	128827	2229786(17.31)	3201737(24.85)
XFree86	12155	171725(14.27)	277254(22.81)
total	498460	8476950(13.77)	10768664(21.20)

7. 翻訳実験

本章では作成した日英対訳コーパスを学習データに統計的機械翻訳で翻訳実験を行った．評価方法には BLEU[6] を用いた．統計的機械翻訳のデコーダには Moses[(a)] を使用した．また，学習には Moses のツールキットを使用した．単語アライメントツールには GIZA++[(b)]，N-gram 言語モデルツールには SRILM[(c)] を使用した．本稿では単語 5-gram の言語モデルを作成した．翻訳モデル作成ツールには Moses の訓練スクリプトを使用した．チューニングには Minimum Error Rate Training（MERT）という手法[7] を用いた．本稿での MERT は BLEU のスコアが最大となるパラメータ値を選択する．テストデータ，開発用データは JF コーパスから 500 文抽出した．

本稿ではいくつか条件を変えて翻訳実験を行った．

1. 配布用の日英対訳コーパスの全てを学習データに使用した場合と JF コーパスのみ訓練データに使用した場合の翻訳結果を比較
2. 実験 1 で作成した 2 つの言語モデルを線形補間
3. 実験 1 で作成した 2 つの翻訳モデルを同時使用
4. 実験 2 で作成した言語モデルで実験 3 と同様に 2 つの翻訳モデルを使用

7.1 実 験 1

作成した全ての対訳文を学習データに使用した場合と JF の対訳文のみ学習データに用いた場合の翻訳実験の評価結果を**表 4** に示す．

表 4 全ての対訳文を学習に使用
Table 4 Results for using all sentences and JF sentences

	BLEU score no MERT	BLEU score with MERT
All	35.19	37.38
JF	40.13	40.02

学習に使った文数が少ないにもかかわらず，JF の対訳文のみを学習に使用した場合のほうが，BLEU 値が高くなった．

7.2 実 験 2

実験 2 では JF の対訳文で作成した言語モデルを全ての対訳文で作成した言語モデルで線形補間した．重みとして 0.1, 0.3, ..., 0.9 を付加した．翻訳モデルは全ての対訳文で作成したもの使用した．線形補間した言語モデルを使用した時の評価結果を**表 5** に示す．weight は重みである．

表 5 言語モデルを補間
Table 5 Results for interpolation of language models

weight	BLEU score no MERT	BLEU score with MERT
0.1	36.41	38.40
0.3	37.42	39.30
0.5	38.31	38.92
0.7	39.56	40.07
0.9	40.08	42.53

言語モデルを補間することで表 4 の JF の BLEU 値を上回った．

7.3 実 験 3

実験 3 では実験 1 で作成した 2 つの翻訳モデルを同時に使用した場合の翻訳実験を行った．翻訳モデルとは reordering-table と phrase-table を指す．言語モデルは実験 1 の全ての対訳文から作成したものを使用した．**表 6** に評価結果を示す．

翻訳モデルを複数用いることで表 4 の JF の BLEU 値を上回った．

7.4 実 験 4

実験 4 では補間された言語モデルと 2 つの翻訳モデルを用いて翻訳実験を行った．**表 7** に評価結果を示す．

Vol.2009-NL-191 No.17
Vol.2009-SLP-76 No.17
2009/5/22

表 6 2 つの翻訳モデルを使用
Table 6 Results for multiple Translation models

BLEU score no MERT	BLEU score with MERT
20.94	41.26

表 7 補間された言語モデルで翻訳モデルを複数使用
Table 7 Results for interpolation LM and multiple TMs

BLEU score no MERT	BLEU score with MERT
37.36	44.36

最も BLEU 値が高くなることを表 7 で確認した．条件を変えることで全ての対訳文を使用したほうが翻訳精度が向上する．

8. 考　　察

8.1 文の対応付け

文の対応付けの精度は全般的に 80%程度であり，その中で文間対応が 1 対 1 なものは 90%だった．対応付けが失敗している例を個別にみると原因として以下のものがあった．

(1) 翻訳文書にある原文書にない表現（翻訳者名や訳注など）が含まれる．

(2) 改行位置が間違っている．

これらを解決することにより，更に対応付け精度が向上すると考える．

8.2 翻 訳 実 験

本稿の実験では，テスト文としては，対応付けスコアの高い 500 文を利用した．この理由は，対応付けスコアの低い文は，対応付けが正しくない可能性があるため，翻訳精度を測定するには不向きだからである．これら上位の対訳文については，BLEU 値が最高で 44.36 と比較的高かった．また，著者らが翻訳結果をみた場合でも，比較的良い訳が得られていた．

ただし，上位の対訳文は，文長が他と比べて短いなど，全体の対訳文の性質を十分に反映していない可能性がある．そのため，今後は，全体の対訳文から抽出した文を人手でクリーニングした文をテストに使うなどして，より代表性の高い対訳文でテストしたい．

9. 結　　論

我々は Web 上のオープンライセンスマニュアルを収集し，文の対応付けを行うことで日英対訳コーパスを構築した．対訳文の総数は約 50 万文となった．構築した対訳コーパスを用いて翻訳実験を行った．実験結果は，更に検討が必要であるが，有望であった．

日英対訳コーパスの存在は貴重である．我々はこのコーパスを公開し，多くの人に使用してもらいたいと考えている．

10. データの公開

構築した日英対訳コーパスは以下の URL で公開する．

http://www2.nict.go.jp/x/x161/members/mutiyama/manual/index.html

11. 使用した言語資源及びツール

(a) デコーダ「Moses」,http://www.statm.org/moses/

(b) アライメント「GIZA++」, http://www.fjoch.com/GIZA++.html

(c) 言語モデル「SRILM」, http://www.speech.sri.com/projects/srilm/

参 考 文 献

1) 那須川哲哉,Danial Andrade, 海野裕也, 村松祐希, 山本和英. 言語横断テキストマイニングのための翻訳対抽出. 言語処理学会第 15 回年次大会, pp.108-111, 2009

2) Masao Utiyama and Hitoshi Isahara. Reliable Measures for Aligning Japanese-English News Articles and Sentences. *Annual meeting of Association for Computational Linguistics*, pp.72-79, 2003.

3) Jörg Tiedemann,Lars Nygaard. The OPUS corpus - parallel and free. *The International Conference on Language Resources and Evaluation*, pp.93-96, 2004.

4) Philipp Koehn. Europarl: A Parallel Corpus for Statistical Machine Translation. *Machine Translation Summit*, pp.79-86, 2005.

5) Masao Utiyama and Hitoshi Isahara. A Japanese-English Patent Parallel Corpus. *Machine Translation summit*, pp.475-482, 2007.

6) Kishore Papinei, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU:a method for automatic evaluation of machine translation. *Annual meeting of Association for Computational Linguistics*, pp.311-318, 2002.

7) Franz Josef Och. Minimum error rate training in statistical machine translation. *Annual meeting of Association for Computational Linguistics*, pp.160-167, 2003.