# 卒業論文

サブワードに基づくニューラル機械翻訳の未知語置き換え

指導教官　　村上 陽平 准教授
立命館大学　情報理工学部
情報コミュニケーション　5 回生
2600160224-0

高田　雅央

2020 年度（秋学期）卒業研究 3（CH）
令和 3 年 2 月 1 日

# サブワードに基づくニューラル機械翻訳の未知語置き換え

高田　雅央

**内容梗概**

グローバル化が進み，英語などの主要な言語のみならず多様な言語が身近で使われることが増えてきている．その結果，機械翻訳の需要はますます高まっており，その正確性やコストパフォーマンスなどを高めていくことが大きな課題となっている．このような課題を解決するために用いられている機械翻訳の手法としてニューラル機械翻訳（NMT）がある．NMT はひとつひとつの単語を訳していく従来の機械翻訳とは異なり，入力文そのものを最小単位とし翻訳していくため，文脈やニュアンスなどを考慮することが可能である．これにより従来の機械翻訳に比べ，語順，構文などのエラーが発生しにくく，正確な翻訳が可能になり，機械翻訳の品質を大きく向上させている．

しかしながら，NMT には利用頻度の少ない専門用語（未知語）を翻訳することが困難であるという欠点が存在する．この問題を解決するために未知語を類義語に置き換えて翻訳する手法が提案されているが，未知語を類義語に置き換えるには未知語が含まれているモノリンガルコーパスを準備しなければならず，未知語ごとにそれらを用意するのは大きな負担になる．

そこで，本研究ではこのようなコーパスを用意することなく，未知語を複数の既知のサブワードに帰着し類義語に置き換える手法を提案する．具体的には未知語をサブワードに分割し，分けられたサブワードから得られたベクトルを合成した新しいベクトルの分散表現から類義語を導き出す．本手法の実現にあたり，取り組むべき課題は以下の３点である．

1. **専門用語の分割**

未知語は既存の用語の複数語の場合もあれば，そうではなく既存の用語の一部を合成して構成されている場合もある．さらに，新語から構成される場合もあるため，辞書に基づく分割だけでなく，既存の大量のコーパスから分割モデルを学習させる必要がある．

2. **サブワードの合成**

未知語の類義語を得るには，未知語に分割したサブワードから生成される分散表現のベクトルを合成し，未知語の擬似的な分散表現を構築する必要がある．

3. **翻訳妥当性の検証**

専門用語部分を上記の方法で獲得した類義語で置き換えた文章の翻訳結果が妥当であるのか検証しなければならない．また，この方法の翻訳精度が通常のNMTと比較して向上してるのかも検証していく必要がある．

一つ目の課題に対しては，専門用語を辞書に基づく形態素解析（mecab）と教師なしの分割学習によって構築された分かち書き器（sentencepiece）を併用しサブワードに分割する．二つ目の課題に対しては，分割されたそれぞれのサブワードから分散表現のベクトルを獲得し，その平均値から類義語を獲得する．

三つ目の課題に対しては，Wikipedia 日英京都関連文書対訳コーパスと日英京都関連対訳用語集を用いて評価を行う．具体的には，前者のコーパスから後者の用語集の専門用語を含む文を抽出し，専門用語部分を提案手法で獲得した類義語で置き換え，その翻訳の妥当性を検証する．評価指標には，対訳コーパスの人手で翻訳された英訳を用いて算出したBLUEスコアを用いる．

本研究の貢献は以下の通りである．

1. **専門用語の分割**

辞書に基づく形態素解析による分割と，大規模コーパスから学習した分かち書き器のハイブリッドの分割手法を考案した．本手法により，日英京都関連対訳用語集内の 100 語のうち 90 語の分割に成功し，mecab での分割は 74 語で，sentencepiece での分割は 16 語であった．

2. **サブワードの合成**

サブワードに分割された専門用語 100 語のうち 90 語のサブワードの分散表現への変換に成功し，90 語の類義語の取得に成功した．90 語の類義語のうち，妥当性があったものは 44 語であった．

3. **翻訳妥当性の検証**

専門用語を獲得された類義語に置き換えて翻訳し，類義語の翻訳部分を日英京都関連対訳用語集内の訳語に再度置き換えて翻訳文を生成した．Wikipedia 日英京都関連文書対訳コーパスの訳文を用いて計算した BLEU 値は 0.21 で，既存の Google 翻訳と比較して 1%精度が向上した．一方，人手の評価では，適切さが 2.7%，流ちょうさが 2.3%低下した．

# Subword-based Replacement of Unknown Words for Neural Machine Translation

Masahiro Takada

**Abstract**

The demand for machine translation is increasing and improving its accuracy and cost performance is a major issue. Neural machine translation is one of the methods for the issue. NMT can consider context and nuances. From this, NMT does not generate syntax errors, NMT can do accurate translations. As a result, NMT has greatly improved the quality of machine translation.

However, NMT is difficult to translate unknown words. To solve this problem, a method of replacing unknown words with synonyms has been proposed. However, that way you have to prepare a monolingual corpus. It is a big burden to prepare them.

Therefore, in this study, without preparing such a corpus, we propose a method to reduce unknown words to multiple known subwords and replace them with synonyms. There are three issues to be addressed in realizing this method.

1. **Division of unknown words**
   Unknown words may consist of multiple terms and it may be composed by synthesizing some of the terms. In addition, it may consist of new words, It is necessary to train the division model from a large number of existing corpora, not only Divide based on the dictionary.
2. **Subword composition**
   In order to obtain synonyms for unknown words, it is necessary to synthesize a vector of distributed expressions generated from subwords divided into unknown words to construct a pseudo distributed expression for unknown words.
3. **Verification of translation validity**
   It is necessary to verify whether the translation result of the sentence in which the technical term part is replaced with the synonyms obtained by the above method is valid. Also, it is verified whether the translation accuracy of this method is improved compared with the normal NMT. There is a need to continue to.

For the first issue, use mecab and sentencepiece together and divide into subwords. For the second issue, Get a vector of distributed representations from subwords and get synonyms from the mean of subwaord's vector. For the third issue, evaluated using the Wikipedia Japanese-English Kyoto-related document bilingual corpus and the Japanese-English Kyoto-related bilingual glossary. The BLUE score calculated using the English translation manually translated by the bilingual corpus is used as the evaluation index. The contributions of this research are as follows.

**[1] Division of unknown words**

We devised a hybrid division method of division by morphological analysis based on a dictionary and a division writer learned from a large-scale corpus. By this method, 90 out of 100 words in the Japanese-English Kyoto-related bilingual glossary were successfully divided, 74 words were divided in mecab, and 14 words were divided in sentence piece.

**[2] Subword composition**

We succeeded in converting 90 of the 100 technical terms divided into subwords into distributed expressions, and succeeded in acquiring 90 synonyms. Of the 90 synonyms, 44 were valid.

**[3] Verification of translation validity**

The technical terms were replaced with the acquired synonyms and translated, and the translated part of the synonyms was replaced with the translated words in the Japanese-English Kyoto-related bilingual glossary to generate the translated text. The BLEU value calculated using the translated text of the Wikipedia Japanese-English Kyoto related document bilingual corpus was 0.21, which was 1% more accurate than the existing Google Translate. On the other hand, in the manual evaluation, the suitability decreased by 2.7% and the fluency decreased by 2.3%.

# ユーザ辞書を用いたニューラル機械翻訳のドメイン適応

# 目次

# 第1章 はじめに

ニューラル機械翻訳（NMT）は，従来の機械翻訳よりもコストパフォーマンスを向上させ，翻訳の品質も大きく向上させている．現在，ニューラル機械翻訳（NMT）として知られている Google が開発した Google ニューラル機械翻訳（GNMT）はニューラルネットワークのディープラーニングによって，人間が翻訳を行ったかのような自然な翻訳を可能にすると同時に，エラーの少なさや正確性においてもその品質を評価されている．また，GNMT の対応言語は 108 言語にまで拡大され，ハワイ語やクルド語など話者数の少ない言語も追加されていることから，その翻訳性能のみならず幅広い場面で活用できることからも注目されている．

統計的機械翻訳（SMT）では，入力文の各単語をマッピングテーブルに基づいてその単語にふさわしい対訳を用意し，学習を行った言語モデルを用いて適切に並び替えることによって翻訳結果を生成し出力する．一方，ニューラル機械翻訳（NMT）では，原文はエンコーダと呼ばれるニューラルネットワークによってベクトル表現に変換され，その後，デコーダと呼ばれる別のニューラルネットワークが翻訳文を生成する．このように SMT ではマッピングテーブルを用いて単語を最小単位とする翻訳を行っていくのに対し，NMT では入力文そのものを翻訳の最小単位とし，入力文に対応する翻訳結果を出力する．また，NMT は翻訳文全体の意味合いや文脈などから翻訳を行うことで従来の SMT など翻訳方法に比べ，より自然な翻訳結果を出力することができる．さらに，NMT はメモリ消費量が SMT に比べはるかに小さいという利点もある．

しかしながら，NMT には，日常的に使われることがあまりない専門用語を含む文章を翻訳することが困難であるということが知られている．NMT では，語彙数が多くなるほど計算量が増え続けるため，語彙数を 30,000～80,000 語程度に制限している．また，NMT の学習は頻繁に表れる語句から優先的に行われるため，出現頻度が少ない専門用語は学習を行えず単一の unk 記号に変換され学習が行われる．したがって，NMT のモデルは，学習コーパスの中にほとんど登場しない専門用語を学習することができない．そのため，専門用語を含む文章の翻訳を行う際，適切な翻訳結果を出力することができない場合がある．それだけでなく，最悪の場合には，NMT の文全体から翻訳結果を出力するという特性から，ノイズとなる専門用語部分の翻訳をせず，結果としてフレーズが訳落ちし

てしまう可能性がある．

また，SMT では，辞書を用いてマッピングテーブルに専門用語を追加することで，専門用語を含む文の翻訳に対応することができた．しかし，NMT にはマッピングテーブルが存在しないため，SMT のように辞書を用いて，専門語を含む文の翻訳に対応するのは困難である．よって，ユーザが NMT をカスタマイズすることも困難であると言える．

そこで，この問題を解決するために専門用語を類義語に置き換えて翻訳を行う手法が存在する．この手法では単語のベクトル分散表現から，専門用語に近い意味を持つ類義語を求め，置き換えが行われる．しかし，学習が行えていない専門用語のベクトル自体が存在しないため，ベクトル分散表現から類義語を導き出すことは基本的に不可能である．また，学習を行うためには新たなコーパスを用意しなければならず，ユーザへの負担も大きくなる．

そこで，本研究では，新しいコーパスを用意することなく，専門用語を複数の既知のサブワードに帰着し類義語に置き換える手法を提案する．このアプローチを実現する上で解決するべき技術課題として以下の 3 つがあげられる．

1. 専門用語の分割
   専門用語は既存の用語の複数語の場合もあれば，そうではなく既存の用語の一部を合成して構成されている場合もある．その場合，分割を行えないことがあるため，形態素解析に基づく分かち書き器の mecab と教師なし学習で分割を行う分かち書き器の sentencepiece の二つの方法を用いる．
2. サブワードの合成
   分割したサブワードから生成される分散表現のベクトルを合成し，専門用語の擬似的な分散表現を構築する必要がある．
3. 翻訳妥当性の検証
   専門用語部分を上記の方法で獲得した類義語で置き換えた文章の翻訳結果が妥当であるのか検証しなければならない．BLUE スコアを用いて妥当性を決めるのではなく，人手でも違和感のない翻訳結果が得られたのか検証する必要がある．

# 第2章 ニューラル機械翻訳のドメイン適応

この章では，ニューラル機械翻訳の構造やプロセス，そしてその特徴や改善点について説明する．また，翻訳の評価に関連し，ドメイン適応が従来の機械翻訳でどのように研究されてきたか説明，そして，ドメイン適応がこれからのニューラル機械翻訳（NMT）にどのように貢献していくのか，その展望について述べていく．

## 2.1 ニューラル機械翻訳

ニューラル機械翻訳（NMT）では，1 つのニューラルネットワークを用意するだけで，学習も翻訳も同じ枠組みで行うことが出来る．つまり，単一のモデルで入力と出力を完結させることが可能である．そして，図 1 に示すように，ニューラル機械翻訳の構造は大きく分けて 3 つに分けられる．まず，1 つ目は入力文を実数値の集合であるベクトル分散表現に符号化するエンコーダ（encoder）である．エンコーダでは，各単語を分散表現と呼ばれる数百次元からなる実数値ベクトルに変換する作業を行う．2 つ目は出力するべき単語を決定する際，どこに注目するのかを調節するアテンション機構（attention mechanism）である．ここでは，エンコードされた入力文と次節で説明するデコーダの内部状態を判断材料として，次の単語を訳出する際に注目すべき箇所を判断し，確率の正規化を行う．そして 3 つ目は符号化された入力文とアテンション情報をもとに出力文を復号化するデコーダ（decoder）である．デコーダでは，コンテキストベクトルと 1 つ前に出力した単語の情報を入力として受け取り，次の単語を出力する．

このように単語のベクトル分散表現だけでなく，単語の前後関係を考慮するニューラル機械翻訳（NMT）の翻訳結果はどれも文章として成立しており，翻訳精度も向上している．しかし，前後の関係を考量するという性質から，対訳のとれない単語があれば，その箇所を飛ばして翻訳を行い，訳抜けが発生し，評価が下がってしまう．これがニューラル機械翻訳（NMT）の抱えている大きな問題の 1 つである．

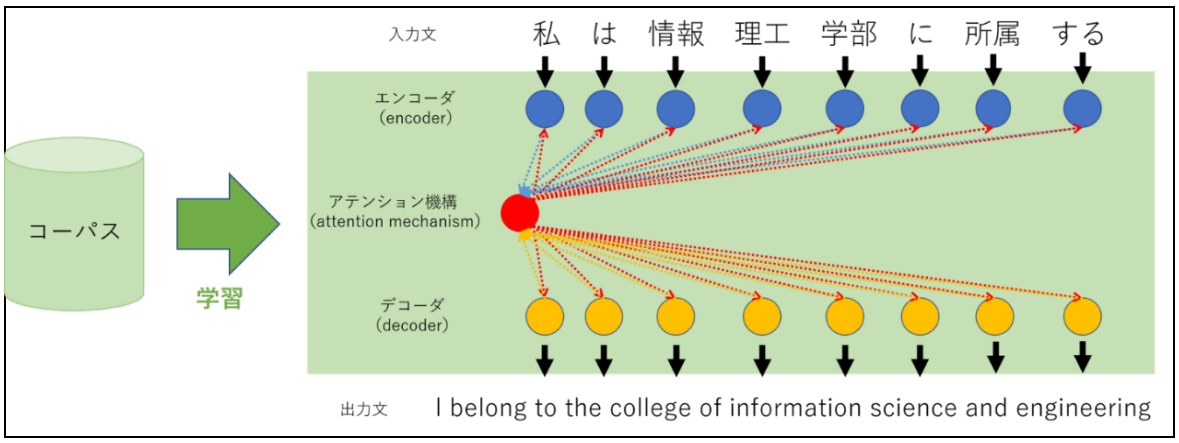


図 1　ニューラル機械翻訳の翻訳方式の概要

### 2.2 ドメイン適応

ドメイン適応とは，ソースデータの分布から、異なる分布をもつターゲットデータで高いパフォーマンスを出すモデルを訓練する手法であり，図 2 のように教師データとテストデータの差異をなくすことがこの研究の目的である．品質の良い大量の教師データを使って学習を行う従来の機械翻訳においても，ドメイン適応が研究されてきた．機械翻訳におけるドメイン適応には，ドメイン外並列コーパスを使用して，ドメイン外パラレルコーパスとドメイン内モノリンガルコーパスのドメイン内翻訳を改善し，動的に訓練データを減らしていき，パフォーマンスを向上させる手法も存在する．しかし，ニューラル機械翻訳（NMT）おいては，ドメイン適応の研究があまりされておらず，学習を行ったモデルを実装したとしても，翻訳品質が低い場合がある．その理由としては，教師データとテストデータの間で差異が生じているからである．機械翻訳では多くの文章や単語を学習する際，出現頻度の多いものから学習が優先されるため，出現頻度の少ない未知語は学習できずにモデルを作成する．このことから，テストデータに未知語が含まれる文章の翻訳を行うと，訳抜けが起こり，教師データとの差異が原因になっている．

この差異をなくし，未知語の学習が行えていないテストデータからでも高いパフォーマンスを実現できるようにすること本研究の目的である．今回は教師データとして「Wikipedia 日英京都関連文書対訳コーパス」にある，「日英京都関連対訳用語集」の対訳文を用意し，テストデータとしては言語グリッドの Google ニューラル機械翻訳（GNMT）で「日英京都関連対訳用語集」の原文の

翻訳を行った翻訳結果を用意する．そして，今回の提案手法ではテストデータの未知語部分を類義語に置き換え，訳抜けをなくし，教師データとの差異をなくすドメイン適応を行う．

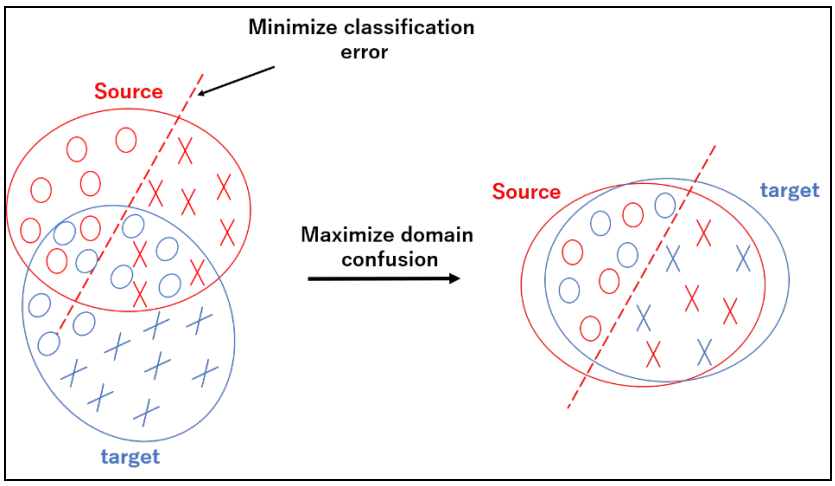


図 2　ドメイン適応の概要

# 第3章 ユーザ辞書を用いた翻訳プロセス

この章では，今回の提案手法である，ユーザ辞書を用いた翻訳プロセスについての説明を行う．

まず未知語が含まれる文章を獲得するために，専門的な用語を含むコーパスを用意する．ここでは「Wikipedia 日英京都関連文書対訳コーパス」を用意する．また，この対訳辞書から未知語と未知語を含む文章を用意する．

次に未知語についての説明する．今回の研究における未知語とは，対訳辞書に載っている対訳と Google ニューラル機械翻訳（GNMT）翻訳結果が異なった単語のことを指す．次に，コーパスから学習を行い，単語を分割するための word2vec のモデルを作成する．このモデルの学習には，さまざまな文章が必要になるため，wikipedia の日本語データを用意する．また，今回は 1 つのモデルで分割できないことを考慮し，形態素解析に基づく分割器である「mecab」と，教師なし学習で分割を行う分割器の「sentencepiece」の 2 つのモデルを用意する．ここまでが，このプロセスを行うための前準備である．

ここからは，本研究のプロセスについての説明を行う．図 3 には，上記のコーパスとモデルを用いて，未知語を含む文章の類義語置き換え，並びに，その翻訳プロセスが示されている．まず，入力文に未知語が含まれているのかの判定を専門用語抽出機を使って行う．判定の結果、未知語が含まれていない場合は直接，ニューラル機械翻訳（NMT）で翻訳して終了する．未知語を含んでいる場合は，その未知語の訳語を辞書サービスで辞書引きを行う．

次に，未知語を類義語に置き換えるために，未知語のサブワードのベクトル分散表現から類義語を求め，文章中の未知語部分を置き換える．今回用意した分割器には mecab と sentencepiece の 2 つがあるが，サブワードへの分割は優先的に mecab を用いて行う．そして，mecab で分割できない場合は，sentencepiece でサブワードへの分割を行う．そして，それぞれの word2vec モデルでサブワードのベクトルを獲得し，その後，ベクトルの合成を行い，類義語を獲得する．また，提案手法の原則として，サブワードに分割して，それらのベクトルを獲得することになっているが，サブワードに分割せずとも，未知語単体からベクトルが得られた場合は，サブワードには分割をせずに，mecab の word2vec モデルを使用して未知語単体から直接，類義語を獲得する．次に獲得した類義語で未知語部分の置換を行う．置換を行った文章を，Google ニューラル機械翻訳（GNMT）

を用いて翻訳文を生成する．以上が提案手法のプロセスである．

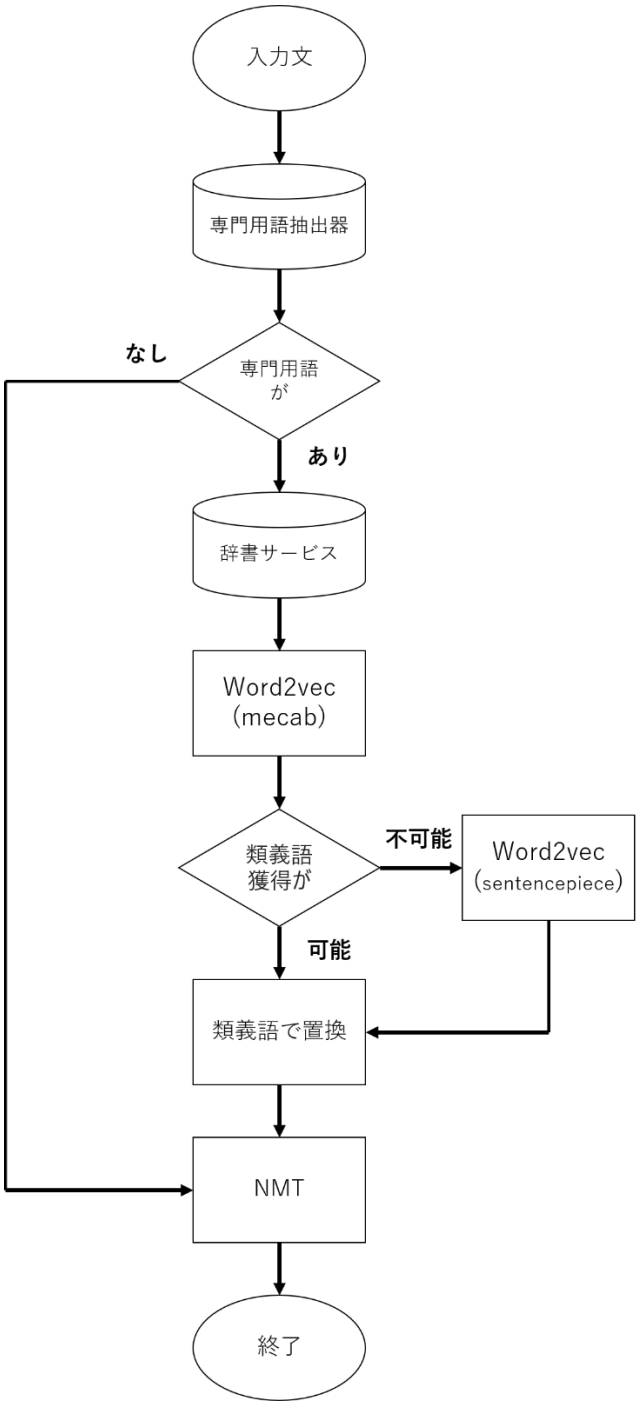


図 3 ユーザ辞書を用いた翻訳のプロセス

# 第4章 未知語の分割

この章では，未知語の分割についての説明を行う．図 4，図 5 は mecab と呼ばれる形態素解析に基づく分割器と snetencepiece と呼ばれる教師無し学習で分割を行う分割器で分割を行った実行結果である．まず図 4 では，mecab と sentencepiece の両方でサブワードの分割が行えていることがわかる．次に，図 5 では sentencepiece ではサブワードへの分割が行えているのにも関わらず，mecab ではサブワードへの分割が行えていない．このことから，mecab 単体と sentencepiece 単体では分割方法の違いから，サブワードが異なったり，分割自体が出来ないことがあるので，今回の実験では，未知語をサブワードに分割できる確率を増やすために，分割器を 2 種類使用する．


図 4


図 5

### 4.1 辞書に基づく分割

今回用いた分割器の 1 つである，「mecab」は形態素解析に基づき単語の分割を行う．形態素解析(Morphological Analysis)とは，自然言語処理分野で主に事前処理として用いられる手法であり，対象となる言語の文法や単語の品詞情報をもとに，文章を形態素(単語が意味を持つ最小の単位)に分解する解析を指す．

図 6 は mecab で word2vec のモデルを作成する過程を示している．まず，大量のテキストが載っている「wikipedia 日本語コーパス」を mecab で分かち書きする．そして出来上がった，wikipedia 分かち書き日本語コーパスから word2vec

の学習を行い，これを word2vec モデル（mecab）とする．

今回のプログラムにおける，パラメータを設定する．今回は[size=200]なので単語ベクトルの次元数は 200 次元である．また，[min_count]は学習を行う単語の出現回数の最低値であり，ここでは出現回数が 20 回未満のものは無視をし，学習を行わないようにしている．[window]は学習に使う前後の単語数である．ここでは対象単語の前後 15 単語の学習を行っている．

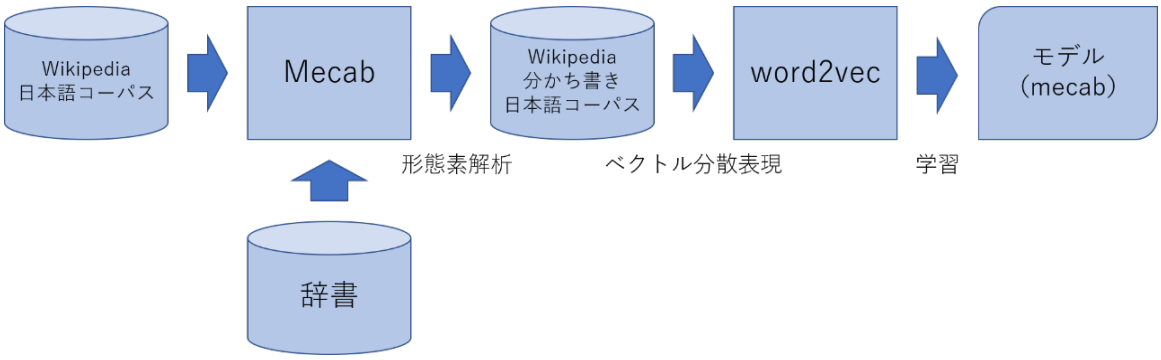


図 6　モデル作成（mecab）

### 4.2 教師無し学習に基づく分割

sentencepiece とは，テキストを単語に分割してくれるトークナイザのことである．Sentencepiece の特徴としては，大規模なテキストデータを短時間で学習することが可能で、形態素解析に比べて扱う語彙数を遥かに小さくすることができることがあげられる．また，Sentencepiece と mecab の違いとしては，sentencepiece は与えられた学習データ（テキスト）から教師なし学習で文字列に分割するという点があげられる．そのことから，教師なし学習で分割を行う，sentencepiece では，文字の分割を行えるようになるための学習を行う必要があり，学習内容やパラメータについての設定を行う必要がある．

図 7 は sentencepiece のモデル作成の図である．ここからはこの図の流れ逃れに沿って，sentencepiece の訓練を行うときの設定やパラメータについての説明を行い，mecab との分割の差異についても述べていく．パラメータについては，まず，[--input]がある．ここには学習データが入ったファイル（ひとつの文が 1 行になっている）が入る．sentencepiece の学習を行うときも，mecab で分かち書きを行った，wikipedia 日本語コーパスを使用する．その理由としては，モデルによって大きな差異が生まれないようにするためである．また，

sentencepieceha は，wikipedia 日本語コーパスそのものからは訓練が行えない．その理由としては，wikipedia 日本語コーパスのデータにはタグが多く含まれており，そのタグが含まれているデータでの訓練が出来ないからである．そこで，wikipedia 日本語コーパスのタグを取り除いた[wiki_removed_doc_tag]という名前のテキストファイルを用意した．[model_prefix]はモデル名であり、モデル名.model とモデル名.vocab が生成される．[--vocab_size]そのままの意味で，語彙数である．wikipedia 日本語コーパスほどのコーパスであれば，32000 語が高い精度が得られる最低値であるので設定も 32000 語に設定した.[--character_coverage]はモデルがカバーする文字の量である．また，このプログラムを実行したところ，データが大きかったため，すべてのデータを学習できなかった．そのため，学習データのサイズを 500 万センテンスに制限した．このような，パラメータ設定を行い，sentencepiece を生成した．

Sentencepiece の生成が完了した後は，モデルの作成を行う．モデル作成の流れは mecab でのものと同じで，単語ベクトルの次元数を 200 次元，単語の出現回数の最低値を 20 回，対象単語の学習を前後 15 単語の行うように設定した．

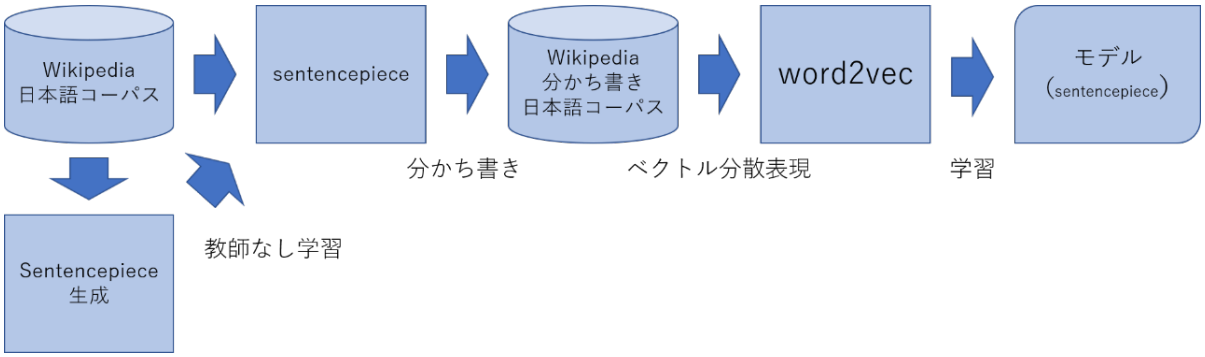


図 7 モデル作成（sentencepiece）

今回の実験を行うに当たって，100 語の未知語の分割を行った．そのうち分割を行えたものは 100 語中 90 語であり，mecab で分割を行った物は 90 語中 74 語であった．また残りの 16 語は sentencepice で分割を行いサブワードを生成した．この結果から，分割できなかった語についての考察や今回用いた分割器の改善点を述べていく．まず，今回分割が行えなかった単語について問題点は二つある．まず一つ目の問題は文字数が少ない単語は分割が難しいということである．

一つ例に出すと，「遊廓」という単語あった場合、これを分ける場合は「遊」と「廓」になる．しかし，このサブワードはどちらも単体ではあまり意味を持たないないため，分割を行わなかった可能性がある．特に mecab は形態素解析に基づいた分割器のため，より分割が困難になる．この文字数の少ない未知語に対するアプローチとしては，もう一つの分割器である sentencepiece のパラメータを変更し，より多くの語彙を学習させることにより，この問題を解決することを検討している．またもう一つの問題として，今回用意した未知語が，一般的な語と見なされるという問題である．先程の遊廓という単語は未知語と用意したが，この単語は比較的に知られている単語であり，未知語としてそもそもふさわしくない場合がある．この場合，このあとの類義語獲得において，未知語単体から類義語を獲得出来，今回の提案手法が翻訳性能を向上させているのかの判断がつきにくくなってしまった．この問題については，テストデータをとってきた段階から人手で未知語の判定を行って，改善していきたいと考えてる．また，今回の分割器の学習を行ったテキストデータは「wikipedia 日本語コーパス」のみであったので，別のテキストデータを用いることで，サブワードがどのように変化するのか検証を行っていきたいと考えている．

# 第5章 サブワードの合成

今回の提案手法のサブワードの合成にあたって，単語の意味の近さをベクトル分散表現で示すことのできる,「Word2vec」を用い，各サブワードのベクトルを求め，それらを合成することで類義語を導出する.

図 8 はサブワードごとのベクトル分散表現を表しており, word2vec のモデル作成時のパラメータ設定についての説明したときと同様に，各サブワードのベクトルの次元数は 200 次元である．このサブワード群のベクトル分散表現から cos 類似度を求め，最も cos 類似度の高い単語を類義語としている．この cos 類似度とは，２つのベクトルがどれだけ同じ方向を向いているかを数値化したもので，cos 類似度が 0 のときには，類似度が低く，1 のときには類似度が高いことを示している．cos 類似度は下記のような式で表すことができる.

$$\cos(\vec{q},\ \vec{d}) = \frac{\vec{q}\cdot\vec{d}}{|\vec{q}||\vec{d}|} = \frac{\vec{q}}{|\vec{q}|}\cdot\frac{\vec{d}}{|\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2}\cdot\sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

$$\cos(\vec{q},\ \vec{d}) = \vec{q}\cdot\vec{d} = \sum_{i=1}^{|V|} q_i d_i$$

この式の分母は, ベクトル q と d の大きさ(ノルム)をそれぞれ掛けたものであり，分子はベクトル q と d の内積になっている.

次に，各サブワードからベクトルを獲得する．プログラムではサブワードのベクトルを model.wv["サブワード"]で獲得する．獲得したベクトルから，類似度の高い単語が獲得できる. 今回の提案手法では，複数のサブワードを用いるため獲得したベクトルの平均を計算し，そこから類義語を獲得する．図 9 は上記のプログラムを実行結果の一例である. 一番上の，実行例は茶経と言う未知語を分割して，類義語を獲得したものである．ここで，茶経の意味とは書物のことであるのだが，得られた類義語では，煎茶という，飲み物を意味する単語を類義語として獲得している．これは，分割結果の「茶」というサブワードの要素が強く出たためである.

```
五
[-3.9397185   2.12252    4.6165442  -1.6339113  -0.18043627 -2.675837
  0.06320617 -1.2840602  -3.312681    2.5507905  -0.9580897  -1.8590086
 -0.30292457  2.0110707  -4.812423    1.3335266  -2.0181723  -3.5464022
      ......       ......       ......       ......       ......       ......
  5.3967404  -3.744996    1.017957   -2.1995933   1.1350121  -1.5436314
 -1.6953205  -0.31040782  0.847443    0.7533174   1.1971139  -0.24654306
  2.8563554   2.0098758   1.0892401  -0.2979981  -3.204053    1.0711583
  3.0279758  -3.056772  ]

老
[ 2.8366635  -2.1001432  -0.29089066 -0.9527352  -3.2941544   0.5741435
  0.8656262   1.5609764  -0.76358217  3.4311965   1.3043519   1.1624463
 -1.0299913   1.321775   -4.2698174   1.2434957   0.20927492 -1.6669303
      ......       ......       ......       ......       ......       ......
  2.9165921  -0.56702155 -4.750387   -1.5371717   4.104979   -5.585199
  2.6036766   2.3841567   1.0045844  -1.582484    5.6351185   0.16135499
 -6.494291    7.5126266  -0.3263871  -1.545107   -6.7279754   0.07615476
  4.2639656   1.7222615 ]

岳
[ 0.11007722  1.247144    0.0515045  -1.3213723   1.2363243  -4.7861705
 -0.61521417 -0.32388616 -0.14013055 4.227027   -0.4775963  -0.43555218
 -3.0889359   2.3424726  -1.6440318  -1.8447586  -1.0765408  -2.5584013
      ......       ......       ......       ......       ......       ......
1.0460086  -0.08974183  4.1136594  -0.86174464  4.0286193  -1.6691662
 0.9937838  -2.612371   -1.528064   -0.3760923   2.3140337  -2.320693
 0.04575034  3.5321913  -0.6723359  -3.60309    -3.7408898   3.5374048
 -3.1163294  -2.1706097 ]
```

図 8　サブワードごとのベクトル分散表現例

**未知語：茶経**
**分割方法：mecab**
**分割結果：茶，経**
**使用モデル：word2vec(mecab)**
**獲得類義語：煎茶**

```
mecab
('煎茶', 0.6168166399002075)
('茶碗', 0.5795539617538452)
('経典', 0.5635163187980652)
('茶器', 0.5627354383468628)
('玉露', 0.55882728099823)
('茶杓', 0.5584231615066528)
('茶の湯', 0.5582609176635742)
('煎', 0.556849479675293)
('経文', 0.5433258414268494)
('酒器', 0.5428525805473328)
```

**未知語：悲田院**
**分割方法：sentencepiece**
**分割結果：悲，田，院**
**使用モデル：word2vec(sentencepiece)**
**獲得類義語：寺**

```
sentencepiece
('寺', 0.6114177107810974)
('寺の', 0.5346105098724365)
('慈', 0.5036659240722656)
('徳', 0.49488410353660583)
('宗', 0.4922068417072296)
('翁', 0.48909056186676025)
('譽', 0.4858201742172241)
('小野', 0.4822464883327484)
('仁', 0.46423470973968506)
('墓所', 0.4629804491996765)
```

**未知語：遊廓**
**分割方法：未分割**
**分割結果：遊廓**
**使用モデル：word2vec(mecab)**
**獲得類義語：遊郭**

```
mecab
('遊郭', 0.9114680290222168)
('遊里', 0.7635488510131836)
('遊女', 0.729992151260376)
('妓楼', 0.7294120788574219)
('花柳界', 0.7080562710762024)
('芸者', 0.6897152662277222)
('娼妓', 0.6816051006317139)
('芸妓', 0.6809511780738831)
('廓', 0.6424344778060913)
('料亭', 0.6421363353729248)
```

図 9　類義語獲得の実行例

上の図 9 は，word2vec を用いてサブワードから類義語を獲得した実行例であ

る．一番上の実行例では，mecab で分割を行ったサブワードから，その mecab で学習を行った word2vec を用いて類義語か獲得を行った結果である．未知語は茶経という書物を意味するものである．結果として獲得した類義語は煎茶という飲み物に関係する単語が導き出された．つまり，この一番上の結果から，正しい類義語獲得が行われなかったことになる．ここから，考察できることは，サブワードの「茶」という語の影響が強く出たためだと考えられる．次に，2 番目の実行例である．2 番目の実行例は sentencepiece で分割を行ったサブワードから，sentencepiece で学習を行った word2vec を用いて類義語か獲得を行った結果である．未知語は，悲田院と呼ばれる，仏教の施設という意味で，建物に関する単語である．ここで期待する結果としては，建物、特に仏教関連の建物が類義語として獲得できることである．実際に提案手法で類義語獲得を行った結果，獲得できた類義語は寺であり，これも仏教に関する施設や建物のことを指しているため，今回の提案手法の成功例と言える．ここで，関連性の高い類義語が獲得できたのは，サブワードの「院」とい文字が，強く影響を及ぼしたためという考察できる．1 番目の 2 番目の実行結果から，サブワードの影響が強く出る箇所によって，類義語獲得がうまくいく場合やうまくいかない場合があるのでないかということが考えられる．最後に，3 番目の実行例の説明を行う．分割を行わずに，類義語が獲得できた結果である．使用したモデルは mecab のものを用いた．ここでは，遊廓が遊郭という単語に置き換わっており，一見類義語獲得に成功しているように思えるが，未知語をサブワードの分割できていない時点で，この提案手法の有用性は証明できておらず，結果として，類義語獲得は失敗と言うことになる．

以上のことから，サブワードに分割する手法や，サブワードのベクトル合成方法を変えることで，獲得出来る類義語がどのように変わり，その結果がどの程度改善できたのか検証行う必要がある．

# 第6章 評価

今回の提案手法であるサブワードに基づくニューラル機械翻訳の未知語置き換えの翻訳精度の評価は，BLEU スコアによる評価と人手による評価の 2 つの方法で行う．

まずは，BLEU スコアについての説明を行う．BLEU スコアは、現在最も広く使用されている機械翻訳の評価方法である．この評価方法の前提は，「プロの翻訳者の訳と近ければ近いほどその機械翻訳の精度は高い」というものである．しかし，今回の研究において，プロの翻訳者は Wikipedia 日英京都関連文書対訳コーパスに収録されている対訳であり，提案手法の翻訳文と従来手法の翻訳文のどちらがその対訳に近づいたのかを比較していく．

下記の数式は BLUE スコアを求めるものである．BLEU スコアは，$N-gram$適合率で評価を行う．$N-gram$とは「隣り合う連続した N 文字」という意味で，BLEU スコアにおいては通常は N=4 として計算される．また，$W_n$は$N-gram$の重みである．

$$BLEU = BPexpW_n \sum_{n=1}^{N} (\log_e P_n)$$

$$W_n = \frac{1}{N}$$

$$P_n = \frac{\sum_i \text{出力文中}i\text{と参照文}i\text{で一致した}N-gram\text{数}}{\sum_i \text{出力文中}i\text{の中の全}N-gram\text{数}}$$

この BLEU スコアを用いて行った．方法は提案手法の BLEU スコアの平均値と従来手法の BLEU スコアの平均値を比較し，その優劣を決定する．

提案手法の母集団の平均の推定値 = 0.2114623173356638
従来手法の母集団の平均の推定値 = 0.20578456368625225
提案手法の母集団の標準偏差の推定値(不偏標準偏差)= 0.19658026995751368
従来手法の母集団の標準偏差の推定値（不偏標準偏差）= 0.1910219727054298
スチューデントの t 検定
提案手法 p 値 = 0.038
従来手法 p 値 = 0.041

BLEU スコアの平均を計算したところ，上記のような結果が得られた．この結果から提案手法の方が，評価値が高くなっていることがわかる．

ここで,この評価結果の差に有意性があるのか判断するために,t 検定を行う．今回は異なる手法から翻訳文を作成しているので，対応なしの t 検定を行う．しかし，t 検定を行ったところ，正規性が確認されなかったため，等分散性を確かめた．結果から BLEU スコアに等分散性が見られたため，スチューデントの t 検定を行うことにした．対応なしのスチューデントの t 検定は，2 つの標本の平均値に有意差がないこと帰無仮説とし，現在の問題設定では，これが棄却されることを期待している．結果は「p=0.038」と「p=0.041」であり，ともに平均値に有意差がないこと帰無仮説が棄却されたため，今回の提案手法と従来手法の平均値に有意差があるということがわかった．

次に人手での翻訳の評価結果について述べる．今回の評価方法として，

{原文　　対訳文　　翻訳文 A　　翻訳文 B}

というフォーマットの評価シートを用意する．この時，原文と対訳文は今回使用した日英京都関連対訳辞書で用意されているものを使用した．そして翻訳文 A，および翻訳文 B には，今回の提案手法で翻訳した翻訳文と従来手法の翻訳文をランダムに入れ，評価者 3 名にバイアスがかからないようにした．下記に示す表 1 提案手法を人手で評価したものである．評価内容に関しては，3 名の評価者に各翻訳文で「妥当性」と「流暢さ」の 2 つの項目で 5 段階評価を行って貰った．そして，3 名それぞれの「妥当性」と「流暢さ」の評価値の平均値を計算し，提案手法のの評価値としている．同様に，

表 2 には従来手法での評価値の一部を載せている．

| |
|---|
| 提案手法の Adequacy = 3.89 |
| 従来手法の Adequacy = 3.94 |
| 提案手法の Fluency = 3.93 |
| 従来手法の Fluency = 3.95 |
| 提案手法の合計の平均点 = 7.83 |
| 従来手法の合計の平均点 = 7.89 |
| 2 群の平均の差 = 0.06 |

次に，評価結果は上記のようなものになった．このことから，人手での翻訳文の評価は従来手法のほうが高いということが言える．

表 1　提案手法の平均評価

| 原文 | 参照文 | 提案手法 | Adequacy | Fluency |
|---|---|---|---|---|
| 落慶とは、寺社などの新築、また修理の完成を祝うことである。 | Rakkei (落慶) refers to celebration of new construction or the completion of repairs to temples and shrines. | the celebration of the construction of a temple is celebrating the completion of new construction or repair of temples and shrines. | 3 | 2.7 |
| 寺請証文は、江戸時代の寺請制度において、寺院が檀家に対して自己の檀家であることを証明するために発行した文書のこと。 | Terauke shomon was a certificate issued by Buddhist temples to danka (supporter of a Buddhist temple) in order to prove that they were actually | A terauke shomon is a document issued in the temple contract system of the Edo period to prove that the temple is a Danke of its own. | 2.7 | 3 |
| 精霊棚は日本の習俗的行事お盆において先祖、精霊を迎えるための棚。 | Shoryodana is a shelf placed to welcome the ancestors and spirits in the Bon festival, which is a conventional event in Japan. | Shoryodana is a shelf for welcoming ancestors and spirits at the Japanese customary event Obon. | 1.7 | 3.7 |
| 大手門とは、日本の城郭における内部二の丸または、三の丸などの曲輪へ通じる虎口に設けられた門。 | An ote-mon gate is a gate constructed at the most important entrance of a Japanese castle that leads to kuruwa (walls of a castle) such as a | Otte-mon Gate is the gate at the tiger's mouth leading to the inner circle of Ninomaru or Sannomaru in the Japanese castle | 3 | 3.3 |
| 書院造は、日本の室町時代中期以降に成立した住宅の様式である。 | Shoin-zukuri is one of the Japanese residential architectural styles which were established after the middle of the Muromachi Period. | Shoin-zukuri style is a style of housing established after the middle of Muromachi era in Japan. | 5 | 4 |

表 2　従来手法の平均評価

| 原文 | 参照文 | 従来手法 | Adequacy | Fluency |
|---|---|---|---|---|
| 落慶とは、寺社などの新築、また修理の完成を祝うことである。 | Rakkei (落慶) refers to celebration of new construction or the completion of repairs to temples and shrines. | The celebration of the construction of a temple is to celebrate the completion of new construction and repair of temples and shrines. | 3 | 3 |
| 寺請証文は、江戸時代の寺請制度において、寺院が檀家に対して自己の檀家であることを証明するために発行した文書のこと。 | Terauke shomon was a certificate issued by Buddhist temples to danka (supporter of a Buddhist temple) in order to prove that they were actually | Terauke shomon is a document issued by the temple in the Edo period to prove to the Danka that it is their own Danka. | 2.7 | 3.3 |
| 精霊棚は日本の習俗的行事お盆において先祖、精霊を迎えるための棚。 | Shoryodana is a shelf placed to welcome the ancestors and spirits in the Bon festival, which is a conventional event in Japan. | Shoryodana are shelves for welcoming ancestors and spirits in the traditional Japanese event Obon. | 4 | 3.7 |
| 大手門とは、日本の城郭における内部二の丸または、三の丸などの曲輪へ通じる虎口に設けられた門。 | An ote-mon gate is a gate constructed at the most important entrance of a Japanese castle that leads to kuruwa (walls of a castle) such as a | Otte-mon Gate is a gate provided at the tiger mouth leading to the Kuruwa such as the inner Ninomaru or Sannomaru in a Japanese castle. | 3 | 3 |
| 書院造は、日本の室町時代中期以降に成立した住宅の様式である。 | Shoin-zukuri is one of the Japanese residential architectural styles which were established after the middle of the Muromachi Period. | Shoin-zukuri style is a style of housing established after the middle of the Muromachi period in Japan. | 5 | 4.3 |

このことを踏まえ，提案手法の評価値が従来手法より，性能を落としてしまった原因ついての考察をしていく．今回、提案手法が評価を落としてしまった原因として，考えられるのは，正しい類義語が獲得できていなかったからだと

考える．なぜなら，人手の評価において，提案手法の妥当性と従来手法の妥当性の差が大きいからである．妥当性とは，意味の通じる尺度であることから，提案手法の方が，意味が伝わりづらいと言うことである．そのような考察をしていくと，類義語獲得に問題があったというの自然であり，今回の提案手法の類義語獲得についての見直しを行っていく．

# 第 7 章 おわりに

今回の実験では，未知語のサブワードに基づいたベクトル分散表現から類義語を獲得し，ニューラル機械翻訳（NMT）を用いて置き換え翻訳を行い，翻訳精度が向上するのかについて検証した．

実験の結果から，うまく類義語を獲得し，翻訳結果も従来手法よりも良い評価が得られたものいくつか存在した．しかし，提案手法と従来手法の最終的な翻訳の評価結果や，実験を行っていくうちに気づいたことから，改善するべき点がいくつもあることがわかった．この結果を踏まえて，改善するべき点について述べていく．

[1] テストデータの選び方

今回のテストデータは「Wikipedia 日英京都関連文書対訳コーパス」から無作為に選んだものであるが，文章の構成や文字数に偏りがあるように感じられた．文章構成においては，すべての原文が未知語から始まっていた．同じような構造の文章だけでなく，様々な構造の文をテストデータとして用意し考察を深めていきたい．

[2] サブワードの分割方法

今回の実験を行っている際，分割を行えない未知語が 10 語存在した．その場合，未知語そのものを mecab の word2vec モデルを用いて，類義語獲得を行った．結果として類義語を獲得出来たのは 10 語中 8 語であった．このように分割を行えず，word2vec で類義語を獲得出来ないということがないように，今回用いた mecab と sentencepiece 以外の分割方法を検証することや，sentencepiece そのものの学習語彙数を増やすことで分割できる語句を増やすことが課題である．

[3] サブワードベクトルの合成方法

今回の提案手法では，未知語をサブワードに分けた後に，ベクトルの合成を行って，cos 類似度から類義語を獲得することになっているが，現段階ではベクトルの平均を使って合成している．そうすると，特定のサブワードの影響が強く出過ぎてしまい，明らかにサブワード群から得られたとは思えない，類義語が存在した．重み付けなどを用いることによって，獲得できる類義語がどのように変化するのかについても調べてきたい．

[4] 獲得した類義語の妥当性

今回獲得した類義語のなかには，未知語とは全く関係のないような，意味の遠いものが見受けられた．そこでその獲得した類義語の妥当性について，しっかりとした基準を作り，判断できるようにする．

このほかにも改善するべきところはあるが，この 2 点が，研究を通して最も改善すべきだと感じた点である．

今回は京都観光関連コーパスを用いて，ニューラル機械翻訳の未知語への対応についての研究を行ってきた．しかし，先程も述べたように，この手法には改善点が多くあることがわかった．分割器の学習一つとっても，語彙数や出現回数などのパラメータを変えることで，よりよい結果を得ることが出来ると考えている．また，翻訳の品質だけでなく，コストパフォーマンスやユーザへの負担を考慮したニューラル機械翻訳のカスタマイズについても研究を進めていきたいと考えている．

# 謝辞

本研究を行うにあたり，熱心なご指導，ご助言を賜りました指導教官の村上陽平准教授に深謝申し上げます．また普段からお世話になっている社会知能研究室の皆さまにも感謝の意を表します．

# 参考文献


[1] 中澤 敏明:機械翻訳の新しいパラダイム：ニューラル機械翻訳の原理，情報管理，60巻(2017-2018)5号，p.299-306

[2] 伊部 早紀，松田 源立，山口 和紀:日英ニューラル機械翻訳におけるアテンションを用いた未知語置き換えの手法，自然言語処理，25巻(2018)5号(2018)，p.511-525

[3] 竹林 佑斗，Chenhui Chu，荒瀬 由紀，永田 昌明:ニューラル機械翻訳における単語報酬モデルに基づく対訳辞書の利用，自然言語処理，26巻(2019)4号，p.711-731

[4] 後藤 功雄，田中 英輝:ニューラル機械翻訳での訳抜けした内容の検出，自然言語処理，25巻(2018)5号，p.577-597

[5] 須藤 克仁:ニューラル機械翻訳の進展 −系列変換モデルの進化とその応用，人工知能，34巻(2019)4号，p.437-445