自然発話に頑健な機械翻訳の検討

村上 聡一朗† 松岡 保静† 内田 渉† 儀田 佳徳† 森下 睦§ 平尾 努§ 永田 昌明§
†株式会社 NTT ドコモ §日本電信電話株式会社
{souichirou.murakami.cr,matsuokah,uchidaw,isoday}@nttdocomo.com,
{morishita.makoto,hirao.tsutomu,nagata.masaaki}@lab.ntt.co.jp

1 はじめに

自然発話を対象とした音声翻訳システムは, 一般的に音声認識システム (ASR) と機械翻訳システム (MT) で構成される. MT は, 人手で整形されたクリーンな対訳コーパスを用いて学習が行われる. これは, ユーザの自然発話が前段の ASR によって正しく認識され, それらテキストが後段の MT へそのまま渡されることを想定しているためである. しかし, 音声翻訳システムの実用においては, ASR によって発話が正しく認識されている場合であっても, 発話内容自体に言い淀みや言い直し, フィラー等のノイズが含まれることにより, 後段の MT の翻訳精度が低下する問題がある.

本研究では, ASR によって認識された, 言い淀みや言い直し, フィラー等のノイズを含む自然発話に頑健な機械翻訳手法を提案する. 具体的には, MT の学習コーパスの原言語側に対してノイズ付与を行うことで, ノイズを含む原言語テキストと元の目的言語テキストで構成される擬似対訳コーパスを作成し, それらの対訳コーパスを用いて MT の学習を行う. 予めノイズを与えた原言語テキストを学習させることで, ノイズに対して頑健になることを期待する.

実験では, ノイズ付与を行わないクリーンな対訳コーパスで学習させたベースラインに対して, BLEU が 2.80 向上し, 提案法により言い淀みや言い直し, フィラー等のノイズに対して頑健な翻訳ができることを示した.

2 関連研究

ユーザの自然発話の翻訳を目的とした音声翻訳システムに関して様々な研究が行われている.

Osamura ら [3] は, ASR の候補単語の事後確率で表されたベクトルをニューラル機械翻訳 (NMT) の学習時の入力として用いることで, ASR の曖昧性を考慮し, 音声認識誤りに頑健な音声翻訳手法を提案した. Sperber ら [6] は, ASR の曖昧性を考慮するために, ASR における単語ラティスと各パスのスコアを入力可能にする LatticeLSTM を提案し, 有用性を示した. これらの研究では, 音声認識誤りが含まれる入力文に対して頑健な翻訳手法に取り組んでいる. それに対し本研究では, 音声認識誤りではなく, 人間の自然発話に多々出現する言い淀みや言い直し, フィラー等のノイズを含む自然発話の翻訳タスクに取り組む.

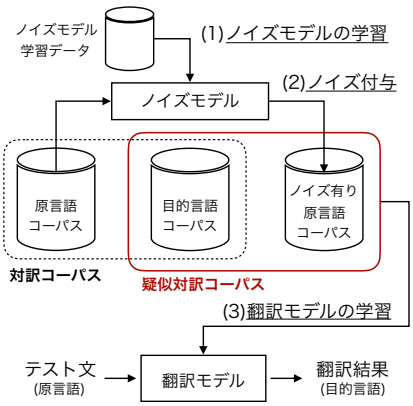

図 1: 提案手法の概要

Sperber ら [7] は, 学習用対訳コーパスの原言語側に疑似的な音声認識誤りを付与するためのノイズモデル用いて擬似対訳コーパスを作成し, NMT を学習することで, NMT が音声認識誤りを含む発話文に対して頑健になることを示した. この研究では, 一様分布や単語のユニグラム確率に基づいたサンプリングにより, ノイズを付与している. それに対し本研究では, 系列ラベリングと単語のユニグラムに基づいたサンプリングによりノイズ付与を行う.

3 提案手法

図 1 に提案手法の概要を示す. 本研究では, ノイズモデルにより既存の対訳コーパスの原言語側に対してノイズを付与することでノイズを含む原言語テキストとオリジナルの目的言語テキストで構成される疑似対訳コーパスを作成し, その疑似対訳コーパスを用いて NMT の学習を行う.

ノイズモデルの学習には, フィラーや言い淀み, 言い直し等がアノテーションされた自然発話の書き起こしコーパスを用いる. 本研究では, 原言語テキストへのノイズ付与を, 入力素性の系列からノイズラベル系列を予測する系列ラベリング問題として考える.

以降では, (1) ノイズモデルの概要, (2) ノイズ付与によるデータ拡張, (3) 疑似対訳コーパスを用いた NMT の学習について詳細に説明する.

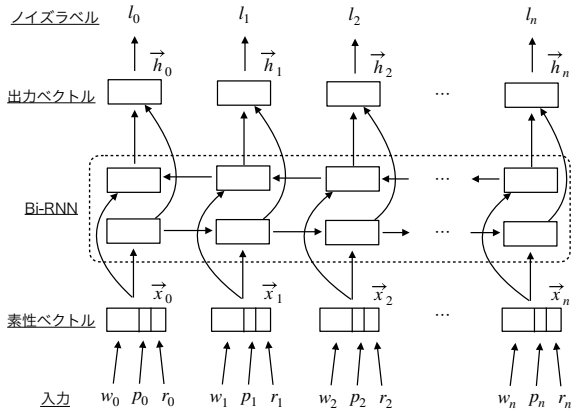

図 2: ノイズモデルの概要

表 1: ノイズラベルのタイプ

<F>	フィラー
<D>	言い淀み, 言い直し
O	ノイズ無し

3.1 ノイズモデル

図 2 にノイズモデルの概要を示す. 対訳コーパスの原言語テキストへのノイズ付与を, 入力テキストの素性ベクトル系列 $\mathbf{x} = (\overrightarrow{x}_0, \overrightarrow{x}_1, \cdots, \overrightarrow{x}_n)$ からノイズラベル系列 $\mathbf{l} = (l_0, l_1, \cdots, l_n)$ を予測する系列ラベリング問題として考える. 本研究では, 品詞タグ付けや固有表現抽出タスク [1] 等で広く用いられている双方向再帰的ニューラルネットワーク (BiRNN) を用いて, ノイズモデルを構築する.

表 1 に提案手法で使用するノイズラベルのタイプを示す. 本研究では, フィラーが付与される要素は <F>, 言い淀み, 言い直しが付与される要素には <D> をノイズラベルとして使用した. また, それ以外のノイズが付与されない要素には, O を付与した.

本研究では, 入力テキストの各トークンの次に適当なノイズラベルを付与するノイズモデルを構築する. ノイズモデルの素性として, 入力テキストの形態素 w_t とその形態素に対応する品詞 p_t, 発音 r_t を用いる. 本研究では, 各素性をベクトル化し, それらを結合することで入力素性ベクトルを作成する.

$$\overrightarrow{x}_t = [\overrightarrow{w}_t; \overrightarrow{p}_t; \overrightarrow{r}_t] \tag{1}$$

ここで, $\overrightarrow{x}_t$, $\overrightarrow{w}_t$, $\overrightarrow{p}_t$, $\overrightarrow{r}_t$ は, それぞれ素性ベクトル, 形態素ベクトル, 品詞ベクトル, 発音ベクトルを表す.

ノイズモデルの学習用発話テキスト「<F えー > それでは会議を <D を > 始め <F あー > ます」を例に, ノイズモデルの学習手順を説明する[1]. まず, 学習用発話テキストからノイズラベルを除去したテキスト「それでは会議を始めます」に対して形態素解析を行い, 形態素系列 $\mathbf{w}$ =(<BOS>, それでは, 会議, を, 始め, ます, <EOS>), 品詞系列 $\mathbf{p}$ =(< BOS >, 接続詞, 名詞, 助詞, 動詞, 助動詞, <EOS>), 発音系列 $\mathbf{w}$ =(< BOS >, ソレデワ, カイギ, ヲ, ハジメ, マス, <EOS>) を取得する[2]. 次に, 式 1 より各タイムステップ t の素性ベクトル $\overrightarrow{x}_t$ を取得し, 素性ベクトル系列 $\mathbf{x}$ を作成する. 続いて, 同学習データのアノテーションされたラベルを用いてノイズラベル系列 $\mathbf{l}$ = (<F>, O, O, <D>, <F>, O, O) を作成する. 最後に, 素性ベクトル系列 $\mathbf{x}$ からノイズラベル系列 $\mathbf{l}$ を予測する系列ラベリングタスクとして BiRNN を学習する. BiRNN は, 入力系列に対する出力系列の予測誤差を用いて, パラメータ学習を行う.

3.2 ノイズ付与

学習済みのノイズモデルを用いて, 既存のクリーンな対訳コーパスの原言語側へノイズ付与を行う. ノイズ付与では, 原言語テキストの素性ベクトル系列に対応するノイズラベル系列を予測し, 対応する形態素の次にノイズラベルを挿入する. 次に, 挿入したノイズラベルをノイズを表す単語へ置換を行い, 最終的な出力であるノイズが付与された原言語テキストを獲得する.

具体的には, まず, ノイズ付与を行う原言語テキストの素性ベクトル系列 $\mathbf{x}$ を学習済み BiRNN へ入力し, 各タイムステップにおけるモデルの出力ベクトル $\overrightarrow{h_t}$ を獲得する. 本研究では, 各タイムステップにおけるラベルを推定する際に, ラベルの事後確率が最大となるものを使わずに, 出力ベクトル $\overrightarrow{h_t}$ に指数を取った値 $\exp(\overrightarrow{h_t}/\tau)$ で定義される多項分布からサンプリングにより推定ラベル決定する.

$$l_t \sim \exp(\overrightarrow{h_t}/\tau) \tag{2}$$

ここで, l_t はタイムステップ t における推定ラベル, $\overrightarrow{h_t}$ はノイズモデルの出力ベクトル, τ は温度パラメータを表す. 本研究では, ノイズの強弱を制御するために, 温度パラメータ τ を導入した. 温度パラメータ τ の値を大きく ($\tau \to \infty$) すると確率分布は一様分布に近づき, 小さく ($\tau \to 0$) すると最も高い確率のラベルを選択するようになる.

次に, ノイズモデルにより予測したノイズラベル系列をフィラーや言い淀み, 言い直し等のノイズを表す単語へ置換を行う. 本研究では, 各ノイズラベルのタイプに対応する語彙集合 V_{type} からユニグラム確率に基づくサンプリングを行う. 例えば, フィラーのノイ

[1] ここで, 学習用発話データに含まれる「<F えー >」は, 「えー」にフィラーのノイズラベル <F> が付与されていることを示す.

[2] 文頭および文末へのノイズ付与を可能にするために, 各素性系列の先頭と末尾にそれぞれ <BOS>, <EOS> を挿入した.

ズラベル <F> をフィラーを表す単語へ置換する場合,

$$w'_t \sim V_{<F>} \tag{3}$$

により, 置換する単語を決定する. ここで, $V_{<F>}$ はノイズラベル <F> の語彙集合, w'_t はタイムステップ t に挿入されるノイズを表す単語である.

以上により, 原言語テキストの形態素系列 $\mathbf{w} = (w_0, w_1, w_2 \cdots, w_n)$ からノイズを含む系列 $\mathbf{w}' = (w_0, w_1, w'_1, w_2, w'_2 \cdots, w_n)$ を獲得する. 例えば, 入力の形態素系列 (<BOS>, では, 発表, を, 始め, ます, <EOS>) に対する予測ノイズラベル系列 $\mathbf{l}$ が (<F>, <D>, O, <F>, O, O, O) の時, 出力 $\mathbf{w}'$ は, (え, では, は, 発表, を, えーと, 始め, ます) となる.

3.3 翻訳モデルの学習

既存の対訳コーパスの原言語側へノイズ付与を行い, ノイズを含む原言語テキスト $\mathbf{w}'$ とオリジナルの目的言語テキスト $\mathbf{y}$ で構成される疑似対訳 $(\mathbf{w}', \mathbf{y})$ を用いて, 翻訳モデルの学習を行う.

4 実験

提案手法の有用性を確認するために, ノイズが含まれないクリーンな対訳コーパスを用いて学習した翻訳モデルをベースラインとし, 日英翻訳タスクで精度評価を実施した.

4.1 実験設定

データセット ノイズモデルのデータセットとして, 講演ドメインにおける自然発話のフィラーや言い淀み, 言い直しがアノテーションされた日本語話し言葉コーパス (CSJ) [3]の転記テキストを使用した. CSJ の内, 1118 講演, 50 講演をそれぞれ学習, 開発用に用いた.

翻訳モデル用の学習用データセットとして, BTEC[8], GlobalVoice[4], TED[5] KFTT[2], および日英中基本文データ[6]で構成される合計約 100 万文の日英対訳テキストを使用した. 本研究では, フィラーや言い淀み, 言い直し等のノイズが含まれる自然発話に頑健な翻訳を目的としているため, 開発および評価用データセットとして, これらのノイズが書き起こしに含まれている同時通訳データベース (SIDB)[7]を用いた. SIDB の内, 1283 発話, 1421 発話をそれぞれ開発, 評価に使用した[8].

また, 提案手法では, ノイズを付与した対訳コーパスを用いて翻訳モデルを学習するため, ノイズが含まれないクリーンな文に対する翻訳精度がベースラインに比べて低下することが懸念される. そこで実験では, ノイズが含まれない文に対する翻訳精度を評価するために, KFTT のテストセット 1160 文を使用した.

テキストの形態素解析には, MeCab[9]を用いる. MeCab の辞書として, IPA 辞書を使用した.

モデル ノイズモデルは, 2 層の BiRNN で構築した. モデルのハイパーパラメータは, 形態素ベクトルは 200 次元, 品詞ベクトルは 50 次元, 発音ベクトルは 50 次元, 隠れ層は 200 次元, ドロップアウトは 0.2, バッチサイズは 20, 温度パラメータは 0.15 とした. また, モデルパラメータの最適化手法には Adam を使用し, 実装には, PyTorch[10]を用いた.

翻訳モデルは, OpenNMT-py(v0.5.0)[11]を用いて Transformer[9] を構築した. Transformer の Encoder および Decoder はそれぞれ 4 層とした. モデルのハイパーパラメータは, 単語ベクトルは 512 次元, 隠れ層は 512 次元, ドロップアウトは 0.3, Multi-head attention のヘッド数は 8 とした. モデルパラメータの最適化手法には Adam を使用し, 学習ステップは 100,000 とした. 対訳文には subword-nmt[12]を用いて BPE によるサブワード化 [5] を行い, 翻訳モデルの学習には, 日本語, 英語ともに 80 トークン以下の対訳文を使用した. また, 学習時に, 1000 ステップ毎にモデルのチェックポイントを作成し, 学習終了時から直近 16 チェックポイントの平均をとったモデルを評価に使用した.

実験では, ノイズ付与を行わずオリジナルの学習用対訳コーパスで学習した翻訳モデルを $Baseline$, その対訳コーパスに対してノイズ付与を行った疑似対訳コーパスを用いて学習した翻訳モデルを $Noise$, オリジナルの対訳コーパスとノイズ付与した疑似対訳コーパスをまとめたコーパスを用いて学習した翻訳モデルを $Original + Noise$ とした. また, 自然発話に含まれるノイズの位置や種類は, 様々なパターンが考えられることから学習時に 1 つの対訳テキストに対して複数パターンのノイズを学習することで, よりノイズに頑健になることが期待できる. そこで, 1 つの対訳に対して複数パターンのノイズを含む疑似対訳をそれぞれ作成した. 特に, ノイズパターン数の有用性を検証するために, 対訳コーパス中の 1 つの対訳に対して 1, 2, 4 パターンのノイズを付与した疑似対訳コーパスで学

[3] https://pj.ninjal.ac.jp/corpus_center/csj/
[4] http://casmacat.eu/corpus/global-voices.html
[5] https://wit3.fbk.eu/mt.php?release=2017-01-trnted
[6] http://nlp.ist.i.kyoto-u.ac.jp/index.php?日英中基本文データ
[7] http://sidb.jp/
[8] SIDB の日本語発話を翻訳者に依頼し, 対訳化を行った.
[9] http://taku910.github.io/mecab/
[10] https://pytorch.org/
[11] https://github.com/OpenNMT/OpenNMT-py
[12] https://github.com/rsennrich/subword-nmt

表 2: BLEU による評価結果

データセット		SIDB	KFTT
ノイズの有無		有り	無し
モデル	$Baseline$	16.08	20.35
	$Noise_{N=1}$	18.86	**20.48**
	$Noise_{N=2}$	**18.88**	20.40
	$Noise_{N=4}$	18.69	20.25
	$Original + Noise$	18.64	20.33

表 3: $Baseline$ と $Noise_{N=2}$ の出力例

入力文	そして 二 階 の 方 は あのー ちょうしょ 料金 に は 含ま れ て い ない ん です けれども あのー ええーとですね 千 円 で えー 日本食 の お 朝食 が とれ ます
参照訳	the second floor restaurant isn 't included in the room price , but for 1000 yen , you can have a japanese style breakfast .
$Baseline$	and on the second floor , it 's not included in the other – but – well , what a pity , what a japanese breakfast is at a thousand yen .
$Noise_{N=2}$	and the second floor is not included in the price , but you can have japanese breakfast for one thousand yen .

習した翻訳モデルをそれぞれ $Noise_{N=1}$, $Noise_{N=2}$, $Noise_{N=4}$ とした．ここで, N はノイズパターンの数を表す.

評価指標　機械翻訳の評価には, BLEU[4] を使用した.

4.2　実験結果

表 2 に各モデルの BLEU による評価結果を示す．表 2 の BLEU スコアは 4 回実験を行った平均値を表している．また, $Baseline$ と $Noise_{N=2}$ の出力例を表 3 に示す.

表 2 のノイズを含む評価データ SIDB に対する BLEU の比較では, $Noise$ が $Baseline$ や $Original + Noise$ に比べて BLEU が向上することを確認できた．$Noise$ モデルの中では, 1 つの対訳に対して 2 パターンのノイズ付与を行った $Noise_{N=2}$ が最も BLEU が高く, $Baseline$ に比べて 2.80 ポイントの向上が確認できた．また, 1 つの対訳に対するノイズパターン数の検証では, $Noise_{N=1}$ および $Noise_{N=2}$ 間では目立った BLEU スコアの差は見受けられなかったものの, $Noise_{N=4}$ ではわずかにスコアが低下した.

表 2 のノイズを含まない評価データ KFTT に対する BLEU の比較では, ノイズを含む対訳のみを用いて学習した $Noise$ モデルにおいて, ノイズが含まれないクリーンな文に対する翻訳精度の低下は見受けられなかった.

表 3 の $Baseline$ と $Noise_{N=2}$ の出力例の比較では, 言い淀みや言い直し, フィラーを含む入力文に対して, $Baseline$ は “price” の訳抜け等の翻訳誤りをしているが, $Noise_{N=2}$ は概ね正しく翻訳できている．これは, 言い淀みや言い直し, フィラー等のノイズを与えた対訳で学習した $Noise$ モデルが, これらのノイズの影響を受けづらくなっている為だと考えられる．一方で $Baseline$ は, これらのノイズを含めて入力文をそのまま翻訳しようとしたため, 出力が崩れる傾向になっていると考えられる.

5　おわりに

本研究では, 言い淀みや言い直し, フィラー等のノイズを含む自然発話に頑健な翻訳手法を提案した．実験では, 原言語側にノイズを含む対訳コーパスを用いて学習した翻訳モデル $Noise$ がノイズを含まないクリーンな対訳コーパスで学習した翻訳モデル $Baseline$ よりも自然発話に頑健になることを示した.

今後の課題として, 自然発話に含まれる長文の言い間違いを無視しながら正しく翻訳できる翻訳モデルの検討などが考えられる.

参考文献

[1] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. In *Proc. of NAACL-HLT'16*, pp. 260–270, 2016.

[2] Graham Neubig. The Kyoto free translation task. http://www.phontron.com/kftt, 2011.

[3] Kaho Osamura, Takatomo Kano, Sakriani Sakti, Katsuhito Sudoh, and Satoshi Nakamura. Using spoken word posterior features in neural machine translation. In *Proc. of IWSLT'18*, pp. 189–195, 2018.

[4] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proc. of ACL'02*, pp. 311–318, 2002.

[5] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proc. of ACL'16*, pp. 1715–1725, 2016.

[6] Matthias Sperber, Graham Neubig, Jan Niehues, and Alex Waibel. Neural lattice-to-sequence models for uncertain inputs. In *Proc. of EMNLP'17*, pp. 1380–1389, 2017.

[7] Matthias Sperber, Jan Niehues, and Alex Waibel. Toward robust neural machine translation for noisy input sequences. In *Proc. of the IWSLT'17*, pp. 90–96, 2017.

[8] Toshiyuki. Takezawa, Eiichiro. Sumita, Fumiaki. Sugaya, Hirofumi. Yamamoto, and Seiichi. Yamamoto. Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world. In *Proc. of LREC'02*, pp. 147–152, 2002.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. of NIPS'17*, pp. 5998–6008. 2017.