

を対訳コーパスから統計的な情報として学習するものであり、現在でもスタンダードな機械翻訳手法として広く使われている。SMT手法の研究が頭打ちになったころ、全く新しい手法として2014年に登場したのが、ニューラルネットワークを用いた機械翻訳、いわゆるニューラル機械翻訳(NMT)である。深い(深層)ニューラルネットワークは、ここ数年で画像、音声、ロボティクス、自動運転など様々な分野において大きな成果を上げている。機械翻訳を含む自然言語処理も同様にその恩恵を受け、それまでの成果を大きく上回る結果を残している。言語の構造解析の詳細、機械翻訳の具体例については、『AI白書2019』「2.3 自然言語処理」を参照されたい。

(2) 主な技術

ここでは、ルールベース翻訳(RBMT)から、用例に基づく機械翻訳(EBMT)、統計機械翻訳(SMT)、ディープラーニングによる機械翻訳(NMT)への進歩を紹介する。また、言語モデルの応用として音声合成などの技術を紹介する。

RBMT(Rule-Based Machine Translation)

初期に作成された機械翻訳は、人が作成したルールを利用して翻訳を行うものだった^{*36}。これをRBMT(Rule-Based Machine Translation)と呼ぶ。1954年の“Georgetown experiment” [4] では、60以上のロシア語の文章を英語に翻訳することができたが、ルールは6つ、扱える単語は250個だけだった。しかし、コンピュータの発展などにより、精度が上がり、1967年に設立されたSYSTRANは、米国防省や欧州委員会などに機械翻訳した文書を提供した。PCの普及と性能向上により、1980年代には、機械翻訳ソフトウェアとしてPC向けのパッケージが販売されるまでに至る。しかし、RBMTは、人手で整備されたルールに基づくもので、「一向に精度が向上しなかったため、機械翻訳の研究は衰退していった」 [5]。

EBMT(Example-Based Machine Translation)

1980年代末～90年代初期にかけて、長尾らにより提案された用例に基づく機械翻訳(Example Based Machine Translation; EBMT)は、翻訳の事例を大量に集めたコーパスを用いた翻訳手法だ。訳したい原文Sに対し、最も類似する文S' と対訳T' の翻訳実例を検索し、事例の訳文T' に基づいて、訳文Tを構築する。高品質の大型両言語対訳コーパスが必要となり、実現は容易ではなかった。

SMT(Statistical Machine Translation)

1980年代からIBMなどが統計的機械翻訳(Statistical Machine Translation; SMT)の研究に着手する。これは、対訳の例文から「翻訳モデル」を、出力言語の例文から「言語モデル」を、機械学習を使って作成する。文章が入力されたとき、翻訳モデルから推測される訳語、言語モデルから得た訳語に対してスコア付けを行い、これを最大化するような組み合わせを求める。このとき、対訳の学習を単語ではなく、フレーズ(複数単語、句)で行うものをPhrase-Based SMT(PBSMT)と呼ぶ。SMT/PBSMTは、RBMTに対してある程度の精度の高さを見せたが、2つのモデルのトレーニングが過大な処理となること、単語区切り、フレーズテーブルなどを人

*36 文法を機械的に解釈して単語で切り分け、辞書から訳語を引き出し、ターゲット言語の文法に沿って並び替えて翻訳を行う。

手で設計する必要があることなどから、やはり一定以上に精度を上げることが難しかった。

NMT(Neural Machine Translation)

これに対して、ディープラーニングを利用したニューラル機械翻訳(Neural Machine Translation; NMT)は、対訳を学習データとし、単一モデルのトレーニングを行うだけでEnd-to-Endで翻訳モデルを構築できる。NMTは、分散表現(「2.2.2 理解」参照)を利用して、単語列(シーケンス)から単語列を発生させるニューラルネットワークである「seq2seq; sequence-to-sequence」 [6] がベースとなった。

図2-2-18は「I'm hungry」を「私は空腹」に訳す場合を例にとって、seq2seqモデルを説明したものである。まず、入力の最初の単語である「I'm」がベクトル表現(分散表現) $emb_{I'm}$ に変換され、これがさらに行列により変換されたうえで非線形関数を経て、新たなベクトル表現 h^1 に変換される。この h^1 が隠れ層の最初の状態である。次に、同様に「hungry」がベクトル表現 emb_{hungry} に変換され、さらに行列により変換される。この変換結果は、隠れ層 h^1 が行列により変換された結果と足し合わされたうえで、非線形関数を経て、新たなベクトル表現 h^2 に変換される。これが入力文の文末まで繰り返され、得られた隠れ層の最終状態(この場合は h^3) に、入力文の意味が表現されていると考える。出力開始記号[開始]を入力し、先ほどと同様の計算を行う。ただし今度は得られた隠れ層 h^4 から翻訳先の言語の単語とその確率を計算する。確率最大の単語(この場合は「私」)を出力し、この単語をさらに次の入力とする。このように単語を一つずつ生成していき、文末記号が生成されるまで繰り返す。これが基本的なseq2seqモデルの挙動である。

■図2-2-18 sequence-to-sequenceモデル

