

## 翻訳メモリツールの使用時に訳文の文脈対応が 失われていく傾向についての考察

内田 順子

(フリーランス英日翻訳者)

### 1. はじめに

1990年代頃から実務翻訳の現場に姿を現した翻訳メモリツールは急速に普及し、すでに産業翻訳ではなくてはならない存在になっている。50年以上もの歴史があるといわれる機械翻訳(Bowker, 2002: 4)が現在もまだ完全な実用化レベルに至っていない(山田 2009)のとは対照的に、後発の翻訳メモリツールが短期間のうちにここまでの成功を収めることができたのは、人間の翻訳者を置き換える存在として機械を設計するのではなく、機械がいかにか人間の翻訳者をサポートするかに重点が置かれるようになった(Bowker, *ibid.*)からではないだろうか。しかし、翻訳メモリツールの活用がますます定着してきている昨今、翻訳者を支援する立場であるはずの翻訳メモリツールが逆に翻訳者に制限を与え、その結果翻訳品質にさまざまな影響が及んでいることが指摘されるようになった。本稿ではその中でも、翻訳テキストが文どうしのつながりの悪い文章になるという、訳文の文脈対応の問題に焦点を当て、翻訳実務の経験を元にした考察を中心に、訳出、編集、レビューといった各工程を経ても文脈対応の問題が温存されていく傾向があるという実態について論証を試みる。

### 2. 翻訳メモリツールと翻訳品質

#### 2.1 文脈への対応

翻訳メモリツールを使用した場合の翻訳品質については、これまでもさまざまな考察が行われているが、その多くで指摘されているのは、訳文が文章としての流れが悪く読みづらいという、文脈への対応に関する問題である。2008年にアジア太平洋機械翻訳協会(AAMT)が一般ユーザーを対象に行った翻訳メモリに関する調査でも、翻訳メモリツールの改善すべき点として「文脈に対応できない」の回答が圧倒的に多かった(機械翻訳課題調査委員会 2009)。この

---

UCHIDA Naoko, "Observations on the Tendency to Lose Context When Using Translation Memory Tools," *Interpreting and Translation Studies*, No.9, 2009. pages 273-285. © by the Japan Association for Interpreting and Translation Studies

問題は主に、テキストを最大で 1 文単位の分節(セグメント)に区切り、この分節単位で原文と翻訳候補をマッチングさせるという翻訳メモリツールの機能性から、翻訳メモリの利点を最大限に引き出すために翻訳者による翻訳行為も 1 文単位で行われていることに関係している (Bowker, 2002: 117)。「文脈」のことを「文章の流れ」と表現することがあるように、2 文以上から構成されるテキストには必ず「流れ」が存在する。玉置 (2004) は、翻訳技法のアプローチには「原文の思考の流れ」をつかみ、その思考の流れを「訳文の自然な流れ」にそって再構成するという 2 つの傾向が見られると考察した。これに対し、翻訳者が 1 文単位で翻訳行為を行う翻訳メモリツールは、原文が 1 文単位で文脈から切り離されて翻訳者に提示される点<sup>1</sup>で「原文の思考の流れ」の正確な理解の障害となり、さらに、同様に文脈から切り離された状態で翻訳者が訳文を入力するという点、および強制的に原文と同じ順序、同じ数の文章で訳出するという点 (Bowker, loc. cit.) では「訳文の自然な流れ」の再構成の障害となり、翻訳技法において重要視されている 2 つの要素に対しまたげとなる性質を持っていると考えることができる。

## 2.2 訳文の再利用性による影響

文脈への対応に影響を及ぼす翻訳メモリツールのもう一つの性質として、訳文の再利用性がある。翻訳メモリツールは、1 分節単位で作成された訳文を原文とペアで保存し、次回にこれと用語および語順が同一または類似した原文を訳出する際に既存訳を訳文候補として表示することで、翻訳の作業効率を大幅に向上させることを最大の利点としている。このため、翻訳メモリツールを使って翻訳する際、翻訳者は、今訳出している分節が現在の文脈以外の別の文脈で再利用される可能性を意識する必要がある。読みやすいように接続詞を補ったり、代名詞を省略あるいは具体的な名称に置き換えるといった、前後の文との結束性に配慮した訳出は、その訳文が別の文脈で再利用されたとき、必ずしも適切であるとは限らない。したがって、翻訳者は翻訳時に意図的に特定の文脈への依存を極力排除しようとする場合がある。山田 (2008) は英文和訳において訳文の再利用効率への配慮が訳出に影響する現象を具体例を用いて示している。このように、翻訳メモリの再利用性を意識して翻訳者が意図的に再利用しやすい文章に訳出している現象は、「peephole translation」(Heyn, 1998, p. 135)と言われ、結果としてテキストの結束性が弱まり、読みやすさが損なわれるという面で、総合的な翻訳品質への影響が指摘されている (Bowker, 2005)。

訳文の再利用性が翻訳品質に及ぼす影響にはもう一つの側面がある。翻訳メモリに保存されている既存訳の再利用時に起きる、既存訳が優先されるという現象である。Bowker (ibid.) は翻訳専攻の学生を 3 つのグループに分け、同じ時間的制約の下で同じテキストを (A) そのまま翻訳、(B) 翻訳メモリを使って翻訳、(C) B のメモリに意図的に誤訳を混ぜたものを使って翻訳させることで、翻訳メモリ使用時の作業速度と品質のバランスを考察する実験を行った際、C グループではファジーマッチ (既出の原文と類似してはいるが同一ではない原文に対して翻訳候補として提案される既存訳) として提示された誤訳が高い率で修正されていたのに対し、完全一致の既存訳として提示された誤訳についてはほとんどがそのまま訳文に導入されていたという結果を得た。この結果について、翻訳者 (学生) はファジーマッチの候補訳については批判的な目を向けるが、完全一致の候補訳に対しては過剰に信頼してそのまま受け入れてしまう

傾向があり、またこの傾向は時間的制約によってより強まる可能性があるという考察がなされている。

### 2.3 訳文の再利用と文脈対応

翻訳メモリに登録されている既存訳は、先述したように翻訳者が一文単位で前後の文脈を意識しないで訳出していたり、あるいは再利用性を考慮して意図的に文脈への依存を排除して訳出しているため、訳文の1つ1つが独立し、文章として組み合わせるときに全体的な結束性が弱まりがちな訳文の集合になってしまう傾向がある。これに加え、翻訳メモリは同一の翻訳プロジェクトを分業する複数の翻訳者によって共有されることが多いため、翻訳者ごとに異なる用語使いやスタイルが混入するなど、翻訳メモリに蓄積される訳文の結束性の問題をより深刻にしている。こうして蓄積された訳文は再び、前後の文脈から独立した形で翻訳者に提示され、既存訳として優先的に再利用されていく。

Bowker による前述の実験では、B グループで使用され、C グループの使用メモリの元にもなっている翻訳メモリは、複数の翻訳者の手による訳文で構成されており、原文ごとにマッチして表示される訳文と訳文の間で、異なる用語使いやスタイルが使われていた。B グループの翻訳者は、こうした一貫性のない用語使いとスタイルをそのまま各自の訳出に採用していたという。特に用語使いの不備については、B グループの全員が翻訳メモリからそのまま採用していたという点も興味深い。これについて Bowker による詳しい考察はないものの、前述の実験結果でメモリに存在していた誤訳をそのまま採用していたのは、あくまでもCグループの一部の翻訳者であり、全員という例はなかった。

C グループに確認された既存訳から新規訳への誤訳の **negative transfer** (山田 2008) は、時間的制約の下で、翻訳者が既存訳を読まずに取得してしまったことによる誤訳の見落としが原因であることが推察される。仮にもっと時間的余裕があれば、翻訳者は既存の訳文候補を読み、誤訳を認識すれば積極的に修正を加えていたであろう。一方、B グループに確認された既存訳から新規訳への翻訳品質の **negative transfer** は、時間に余裕のない翻訳者が訳文候補として提示された既存訳を読まずに盲目的に取得したことによるものなのか、既存訳に目を通して、用語やスタイル上の不整合に不満を持ちつつも既存訳を優先することを選択した結果なのか、明確な検証が行われていない。したがってこれは推論の域を出ないが、仮に翻訳者が既存訳を読み、前後の文脈との結束性が弱く読み難い文章と認識していたとしても、対になっている原文に対する誤訳でさえなければ、既存訳をあえて修正しない場合が多いのではないだろうか。

一つの原文に対して「基本的に正しい」訳出は一つとは限らず、翻訳者は得られる情報や自らの経験と知識をもとに適切と思う訳出を選択するのであり、その選択に対して責任を負うことになる。一方翻訳メモリは、翻訳者自身の知識や経験とは無関係な所から、訳出候補を提示してくる。しかも実務翻訳では、翻訳案件の発注元が、依頼とともに翻訳メモリを翻訳者に提供し、その使用を指定してくる場合が多い。こうして提供される翻訳メモリは、以前にその案件を手がけた翻訳者によって翻訳され、校閲者によって承認された上で、発注元の資産になっていることを背景とした権威を持っている。Pym (2008) は、このような翻訳メモリから提案される既

存訳を採用することは、翻訳依頼主の要望を優先させることにつながり、また万が一、採用した既存訳に品質上の問題があった場合でも、最初はその訳を提供した翻訳者およびそれを黙認した校閲者の責任にできるという理由から、比較的リスクの低い選択肢であり、翻訳者はそれを選択する傾向にあると指摘している。

### 3 翻訳メモリツールと編集作業

#### 3.1 編集作業の定義

翻訳メモリツールが翻訳品質に及ぼす影響について考察してきたが、翻訳メモリツールそのものは機械だとしても、翻訳メモリに保存されている、翻訳者の新規訳に影響を及ぼす既存訳(山田 2008)は、結局は人間の翻訳者によって作成されている。したがって、翻訳メモリを使用して高品質の翻訳を達成するためには、翻訳メモリに保存される訳文が正確であることが前提条件になる(Bowker, 2005)。翻訳メモリを使用するしないにかかわらず、翻訳全般において、訳出作業の後には、ミスを修正し翻訳品質に磨きをかけるための編集作業が必ず行われる。前述の諸問題を厳密に訳出作業時の問題とするならば、それらに対する対策は編集作業において講じられるものと考えることができる。

実務翻訳における「編集作業」は、それが意味する実際の作業や目的はさまざまで、定義も曖昧なのが実情だが、翻訳者としての経験とインターネットでの議論を元に Wetz (2001) が提案した定義によれば、「編集 (Editing)」とは誤りを訂正し、正確さとスタイルを改善するために、翻訳を修正し「磨く」作業である。訳出作業を手がけた翻訳者自身によって納品前の確認作業として行う場合と、訳出を手がけた者とは別の翻訳者が行う場合があり、いずれも基本的に翻訳者が行っている。「レビュー (Review)」は、翻訳を全般的に検査し、その品質、適合性、修正の必要性について意見を出す作業である。翻訳会社によっては翻訳および編集を手がけた者とはさらに別の翻訳者が担当する場合もあるが、最終的には翻訳依頼主(翻訳会社または翻訳物を実用する企業)によるレビュー工程(QA)が入り、意見の階層順位は翻訳依頼主の方が明らかに上であることから、ここでは総合して翻訳依頼主が行うものと解釈する。

以上をまとめると、実務翻訳における「編集」とは翻訳者が翻訳の作業成果を最善の状態で依頼主に提示するために、または編集作業自体を仕事としてその作業成果を依頼主に提示するために行う作業で、これに対して「レビュー」は翻訳依頼主がそれらの作業成果を評価する作業、と考えることができる。

#### 3.2 編集作業と翻訳メモリツールの関係

翻訳メモリ特有の現象としての結束性の問題は、編集作業時に後回しで処理される(山田 2009)ものとされるが、現実にはそれが実現できているかを知るには、編集工程に翻訳メモリツールがどのように関わっているのかをまず明確にしておく必要がある。翻訳者によるワープロの使用が普及しはじめた頃、多くの翻訳者が編集作業時には翻訳物を一度印刷することで、ワープロの影響から切り離された状態で翻訳の見直しをしていたように、極論すれば、翻訳メモリツールの影響が及ばない状態で編集作業を行うのが最善と考えられる。だが、実務翻訳の場合、翻訳メモリツールを使ったプロジェクトでは編集作業も同じ翻訳メモリツールの環境下で行うこと

が指定されている場合が多いため、編集工程で翻訳メモリツールを使わずに作業することは難しい。

依頼主からツールの使用を指定されて作業する場合は、翻訳メモリも依頼主の資産であり、翻訳物と同様に納品の対象となる<sup>2</sup>。この場合、編集作業で加えるすべての修正・変更は、翻訳ファイル上の訳文だけでなく、翻訳メモリに登録されている訳文にも反映されることが要求される。翻訳メモリと翻訳対象ファイルは、翻訳メモリツールのインターフェイスを使って作業する際は連動しているが、物理的には別個のファイルであるため、片方に変更を加えたとすれば、同じ変更をもう一方にも加えなければ同期されない。翻訳対象ファイルに行った修正・変更を翻訳メモリにも反映させる方法としては、訳出作業時とほぼ同じ要領で翻訳メモリと連動させながら分節単位で編集作業を行うことが一般的である。

もう1つの方法として、「バイリンガルファイル」と呼ばれる、分節ごとに原文と訳文が一組になって表示され(原文は隠し文字になっているので表示/非表示の切り替え可能)、その内容を一括処理で翻訳メモリに読み込ませることが可能な形式(SDL Tradosで翻訳作業中の翻訳ファイルはすべてこの形式)のファイルを、翻訳メモリと連動させることなくMicrosoft Word上で通常のWord文書と同じように編集するやり方も考えられる。この場合、編集時に行った変更内容を翻訳メモリに反映する作業は、翻訳メモリツールの機能(「翻訳/Translate」または「クリーンアップ/Cleanup」)を使って一括処理で行う。この方法は、翻訳メモリツールの影響が及ばない状態で編集できるという利点がある反面、翻訳者が意図しないところで翻訳メモリに深刻な問題を生じさせる可能性がある。

まず、バイリンガルファイルに使用されているさまざまなタグ(書式情報を記録したり、原文と訳文のペアをリンクさせる役割を果たす)を編集時に誤って削除してしまうと、後の工程で必要な処理をかけられなくなる。削除してしまったタグの種類によっては、問題箇所を調べるための検証処理で検出できないため、手作業で検証を行うか、一から翻訳ファイルを作成し直して問題を解決する必要がある。さらに、同一の原文が文書内に繰り返されている場合(Repetitionと呼ばれる)、翻訳メモリを使わずに編集を行うと、以前に同型文が存在したかどうか、どのような訳出を行っていたかを意識せずに修正してしまう危険がある。原文が同一の訳文を各所の文脈に合わせてさまざまに修正してしまうと、翻訳メモリは1つの原文に1つの訳文を一对とすることが前提になっているため、後の処理やメモリ再利用時に問題になることがある(「3-4. 文脈に配慮した編集作業に対する制約」を参照)。

これらの技術的な問題は、編集を行う翻訳者がツールに習熟していれば回避できる場合もある。しかしこれとは別に、編集後に変更内容を翻訳メモリに反映させる処理をかける際、訳文とともに翻訳メモリに登録されるさまざまな付帯情報(次項参照)が更新されてしまうという問題もある。付帯情報には履歴を表示する機能がないため、一度更新されてしまうと、更新前の情報は参照できなくなる。これらの付帯情報は後続の処理や翻訳の再利用時に重要な役割を果たす場合も考えられるため、先述したように翻訳メモリも依頼主の資産であるという立場上、翻訳メモリ内のデータの不用意な改訂を伴う処理を依頼者の承諾を得ずに行うことはできない。

以上の理由から、バイリンガルファイルに直接編集を入力してから翻訳メモリに読み込ませるという編集方法は、翻訳者の一存では採用し難くなっている。したがって、結果的には編集作

業も翻訳メモリツールの環境で行わなければならないのが現状である。

### 3.3 編集作業環境としての翻訳メモリツールの非効率性

このように、翻訳メモリツールが使用された翻訳物の編集作業では、コンテンツが翻訳ファイルと翻訳メモリに分散しているため、翻訳者は翻訳の見直し・修正作業に加え、変更内容のメモリへの同期にも配慮する必要がある。具体的には、訳出作業時とほぼ同じ要領で、翻訳メモリと連動させながら分節単位で編集作業を行うか、翻訳ファイルに直接編集を入力してしまってから、後で変更内容を翻訳メモリに手作業または一括処理で反映させることになる。いずれにせよ、このことは、分節という翻訳単位のとらえ方が、編集作業においても継続していることを意味する。ほとんどの翻訳メモリツールには訳文を原文の元のドキュメントレイアウトで表示できるプレビュー機能があり、翻訳メモリツールのインターフェイス内で編集作業を行う場合も、分節にとらわれずにテキスト全体として訳文を確認することに役立つが、プレビュー表示は翻訳メモリと連動していないため、実際に訳文に修正を加える場合には再び編集画面に戻り、該当する分節を開いて分節単位で作業することになる。翻訳ファイルに直接編集を入力してしまう場合でも、後にその変更内容を翻訳メモリに保存することを考えると、分節の区切りや順序を著しく変えるような修正は行いにくくなる。

さらに、翻訳メモリツールのインターフェイス内で分節単位で編集作業を行うことが重視されるもう一つの動機として、分節の付帯情報の表示がある。翻訳メモリツールには、翻訳メモリと連動する用語ベース(用語のデータベース)機能があり、翻訳対象分節の原文に用語ベースに登録されている単語が含まれていれば、その単語がハイライト表示されるとともに専用の領域に対訳が提示される。訳文テキスト全体での用語の一貫性も必要だが、訳文に採用されている訳語と用語ベースに登録されている指定訳(採用することが指定されている訳)との間に不整合があれば、これもエラーとして編集時の修正対象になることから、この情報は編集作業においても重要な意味を持つ。また、翻訳メモリによって提示される既存訳について、その作成者、作成日、変更者、変更日、属性などの情報も表示され<sup>3</sup>、これらの情報は特に複数の翻訳者による分業において重要な意味を持つ場合がある。しかし、このような一連の付帯情報は編集対象の分節を開く(翻訳メモリと連動した編集対応モードにする)操作を行わないと画面に表示されないため、付帯情報の参照が編集作業に重要な意味を持つ場合には、訳文に修正を加えるかどうかにかかわらず、単に訳文に目を通すだけの目的でも分節を開く必要性が生じることになる。

分節の付帯情報に関して、もう一つ編集作業時に特有の問題がある。翻訳メモリツールを使った翻訳作業時には、翻訳対象の原文と同一または類型の原文が以前に翻訳されている場合、翻訳メモリに存在する既存の訳文が提示されるとともに、その原文と現在翻訳対象の原文との一致率がパーセント値(100%=完全一致、99%~=ファジーマッチ/不完全一致)で表示される。さらにファジーマッチの場合は、既存訳の原文と現在翻訳対象の原文の差異が解析され、具体的にハイライト表示されるので、翻訳者はこの情報を元に差分のみを翻訳して新規の訳文を作成できる。ファジーマッチとして提案された訳文を元に翻訳者が差分を翻訳して翻訳メモリに登録すると、登録と同時に分節の付帯情報が更新され、次回以降にこの分節を開いた

ときは、この訳文は完全一致(100%一致)の訳文として表示されるようになる。付帯情報には履歴を表示する機能がないため、この訳文がファジーマッチの既存訳を元に作成されていること、ファジーマッチの一致率は何パーセントで、既存訳と現訳ではどの部分に差異があったのかなどの情報の表示はなく、最後に翻訳を手がけた(差分のみを翻訳した)翻訳者を作成者とした、現在の原文に100%一致する既存の訳文であるという情報のみが表示される。つまり、翻訳が終了した後、編集時に分節を開いて見るときには、ファジーマッチの既存訳に手を加えただけで作成された訳文も、一から完全に新規で翻訳された訳文も、互いに区別できない状態で表示されるのである。

ファジーマッチは、既存訳に手を加えるだけで訳文を作成できることから、参考訳がなくゼロから訳出する場合と比べて翻訳者の心的負担が少なく、翻訳メモリが作業効率を高める要因の1つであることが検証されている(山田 2009)。編集作業の場合も、以前に一度翻訳および校閲を経て翻訳メモリに入っている既存訳に、新たに必要な改訂を加えただけで作成された訳文を編集する場合は、実質的には改訂部分のみを品質確認すれば良いのであるから、完全に新規翻訳された訳文を編集する場合よりも心的負担が少ないものと考えられる。しかし、ファジーマッチから派生した訳文も、完全に新規翻訳された訳文も、編集時には区別なくすべて同じように表示されるということは、編集者はすべての訳文が新規翻訳であると想定して全文の品質確認を行わなければならない、翻訳時に存在したファジーマッチによる作業効率を編集作業時には期待できないことになる。

以上から、翻訳メモリツールを編集作業の環境として見た場合、(1) 編集対象のコンテンツが複数箇所に分散している、(2) 分節を翻訳単位とする制約が継続している、(3) 必要な付帯情報の表示において編集効率が配慮されていないといった非効率要因が存在することがわかった。こうした非効率性は、翻訳メモリツールが訳出作業で使用されることのみを想定して設計されていることを示唆している。今後の翻訳ツールの機能性向上のためにも、また翻訳ツールの使用を視野に入れた翻訳技法のあり方を考える上でも、研究の余地がある領域であると思われる。

### 3.4 文脈に配慮した編集作業に対する制約

翻訳メモリツールを使って翻訳されたテキストの文脈対応の問題は、翻訳者による翻訳行為が1分節単位で行われていることに大きく関係していることはすでに述べた。翻訳時の問題は編集段階で処理されるべきだが、実務翻訳の場合は編集も翻訳メモリと連動させながら分節単位で行うのが一般的であるため、分節という翻訳単位上の制約が編集作業においても継続していることがわかった。もちろん、これまでに検証されてきた翻訳メモリツール特有の翻訳品質の諸問題について、訳出および編集作業にあたる翻訳者が明確な問題意識を持ち、ツールのレビュー機能を活用するなどして文脈への配慮を心がけていれば、翻訳の質を向上させることは可能である(山田 2009)。ただ、具体的に何をどこまで修正できるかという点では、分節の区切り方や順序の変更を伴う種類の修正には、訳文の再利用率が下がるなどのリスクが伴う。例えば、次のような例が考えられる。

例 1.

<原文> Supported languages include: Java, HTML/JavaScript, XML, and SQL.

<訳文> 対応言語は以下を含みます: Java、HTML/JavaScript、XML、SQL。

<編集候補> Java、HTML/JavaScript、XML、SQL などの言語に対応しています。

翻訳メモリツールの分節区切りは原文のパンクチュエーションを基準に行われ、ほとんどの場合、コロン(:)は分節区切りの対象になる。したがって、上の例の原文は"Supported languages include:" と "Java, HTML/JavaScript, XML, and SQL." という2つの分節に分かれている。<編集候補>の訳文は読みやすく日本語としての流れも自然だが、この訳出のためには、2つの分節を結合するか、順序を置換する必要がある。分節の結合は再利用率に影響し(次回以降も、"Supported languages include:" という分節の次に必ず "Java, HTML/JavaScript, XML, and SQL" という分節が続き、かつ、これらの分節を結合するまで、この訳文は一致しない)、順序の置換はその訳文が別の文脈で採用された場合に誤訳の原因になる。この結果、訳文の再利用の必要がない(同じ文の繰返しがない)ことを確認したり、分節の結合または置換について依頼主の承認を得たりすることができなければ、編集者はこの訳文をそのまま採用するか、語順を変えずに別の訳を考えることになる。

分節による制約以外に、1つの原文に1つの訳文という1対1対応のペアが重視される翻訳メモリ特有の現象(山田 2008)も、文脈に配慮した編集行為に影響を及ぼす場合がある。例えば、次のような例が考えられる。

例 2.

<原文> Yes

<訳文 1> はい

<訳文 2> 有

<訳文 3> 可

<原文> No

<訳文 1> いいえ

<訳文 2> 無

<訳文 3> 不可

<訳文 1> 文脈:     トラブルシューティングの解決手順フローチャート。「問題は解決しましたか?」→「はい」「いいえ」の選択に応じて次に進む手順が決まる。

<訳文 2> 文脈:     製品仕様の表。「無線 LAN」などの機能項目に対して「有」「無」。

<訳文 3> 文脈:     提供サービスのリスト。「延長保証」などの見出しに対して「可」「不可」。

上の例の訳文は、同じマニュアルの別々の章の中に存在するため、すべての訳文は同一の翻訳メモリ内に保存される。翻訳メモリのデフォルトの設定では、原文と訳文のペアを1対1で維持するため、同一の原文に対して既存の訳文と異なる訳文を登録すると、既存の訳文が新たに登録した訳文で上書きされる。したがって、上の例で仮に訳文が上から順(1、2、3の順)に登録された場合、最終的に翻訳メモリには「訳文 3」のみが残り、それ以降は文脈に関わりなく常にこの訳が提示されるようになる。このような場合に対処するため、翻訳メモリには1つの原文に対して複数の訳文を登録することを許可するオプションがあり、1対複数のペアで翻訳メモリ

を作成することも可能ではある。ただし、複数ある訳文候補から特定の文脈に適した訳文を選ぶには翻訳者の介入が必要であり、自動翻訳を行った場合は文脈に関わりなくある訳文が繰り返し取得されてしまうといった問題が起こることがある。このような問題を避けるために用意されている設定を使って1対複数のペアにペナルティ(原文の実際の一致率から差し引かれるポイント)を設定することも可能だが、本来なら完全一致である訳文が事実上完全一致ではなくなることで、訳文の再利用効率が下がる、翻訳料金に影響する(メモリのマッチ率はワードカウントとともに課金の基準にされるため)など新たな影響が生まれるため、翻訳者の一存では判断できないところがある。さらに、1対複数のペアとしてすべての訳文を維持すれば、将来この翻訳メモリが再利用される際、自動翻訳処理あるいは翻訳者の見落としによって、別の文脈用の訳が採用されてしまったり(上の例では<訳文 1>の文脈に<訳文 2>など)、スタイルが混在(Yesは「はい」だが対応するNoは「不可」など)したりするリスクが生じる。したがって、文脈に配慮した複数の訳出を正当化するにはリスクがあまりに大きいと判断された場合には、多少の不自然はあってもすべての文脈に対して誤りにならないような1つの訳が優先されることになる。

翻訳メモリツールの翻訳品質への影響を論じた際に、文脈の流れあるいは文の結束性上の品質に問題があっても情報に誤りはない訳文の場合、翻訳者はこれを優先的に訳出に導入する可能性があることを示したが、この傾向は編集作業においても同じであることがわかる。いずれの場合にも、背景には訳出および編集にあたる翻訳者のリスク回避という意識がある。リスクを負うことに対して報酬(金銭的、象徴的、または社会的)が与えられるのであれば、翻訳者はリスクに対して積極的にになれるという Pym(2008)の指摘は、こうした背景を的確に表している。あからさまな誤訳であれば、それを修正することには「誤情報の伝達を未然に防いだ」という象徴的な報酬や、逆に修正しなければ誤訳したまたは見逃したという責任があるので、翻訳者は積極的にこれを修正する。しかし、文脈対応の問題については、このような報酬または責任の所在に不明なところが多く、実務翻訳者はリスクに対して消極的にならざるを得ないのではないだろうか。さらに翻訳メモリツールは、分節という制約や訳文の再利用性への配慮といった新たな責任を翻訳者に課すことにより、翻訳者が直面するリスクをより大きくしていると言えるのではないか。

## 4. 作業成果の評価

### 4.1 レビュー工程における品質のとらえ方

前述したように、訳出作業および編集作業の作業成果は、レビュー工程において評価されることになる。レビューの結果が悪ければ、翻訳依頼企業は翻訳会社に、翻訳会社は翻訳者に、やり直しの指示や報酬の見直しを求めることになり、逆に結果が良ければ今後の翻訳会社および翻訳者の選考において考慮されることになるので、翻訳会社および翻訳者の仕事の継続性につながる。つまり、レビュー工程は翻訳者への「金銭的、象徴的、または社会的な報酬」において重要な影響力を持っていると考えられる。翻訳者とリスクに関する Pym(2008)の考察にあてはめると、レビュー工程において重視される問題点については、翻訳者は積極的にその改善につとめるが、逆にレビュー工程において問題にされない点については、翻訳者は独自の判断に基づいて干渉することに消極的になると考えることができる。

翻訳を発注した企業内のレビュー(QA)工程がどのように行われているのか、公開されているデータには限りがあるが、自動車やITなど市場が国際的で翻訳需要の高い産業では、翻訳プロジェクトの品質評価について、評価基準が主観的で一貫性がない(Sirena 2004, LISA 2004a, 脇田 2009)、時間と費用がかかる(脇田 *ibid.*)といった問題点が積極的に指摘されており、これらに対処するために品質評価の測定基準を社内規模あるいは業界規模で標準化する試みが進められ、具体的な実践報告が行われている。これは、翻訳物に含まれる「エラー」を分類し、重要度に応じて重みを調節したスコアを設定したものを基準にレビューを行うことで、品質評価の安定と効率化をはかるといえるものである。これらの実践報告を概観してみると、言い回しやスタイル上の問題点はエラー分類から意図的に除外されている(Sirena *ibid.*)か、分類対象にはなっていないでも重要度は軽く設定されている(脇田 *ibid.*)。これは、スタイル上の問題は評価が三者三様に異なる(Sirena *ibid.*; LISA 2004a, 脇田 *ibid.*)上に、修正に時間がかかるのに、明確な成果が得られない(LISA 2004b: 17)と考えられているためである。そのため、検出や修正の基準がわかりやすく、かつユーザーへの情報伝達に明確な影響がある誤訳や訳飛び、否定／肯定や数値の取り違えのような、テキストの情報コンテンツの正確性に重点をおいた品質の定量評価の方が重視されている(Sirena *ibid.*; 脇田 *ibid.*)。

翻訳者は訳文の文脈対応を改善するために、接続詞を補う、代名詞を省略あるいは具体的な名称に置き換える、語順を変える(「3-4. 文脈に配慮した編集作業に対する制約」の例1.)、言い回しを変える(同例2.)などの方策を取ることができるが、こうした具体的な方策の多くは、判断基準が主観的で一貫性がない。しかも、こうした方策を取らなくても原文の意味を忠実に訳出することに直接的な影響はないため、品質評価ではスタイル上の問題として重要視されない可能性が高い。翻訳メモリツールを使用することによって、訳文の再利用性という条件が考慮されると、上述のような訳文の文脈対応に配慮した具体的な方策は、訳文の再利用性を低下させるばかりでなく、別の文脈での再利用時に誤訳の原因となる危険を伴うことになる。訳文の再利用性の低下や誤訳は、検出や修正の基準が明確であり、品質上の問題として重要視される可能性が極めて高い。このように、文脈対応に配慮した翻訳者の努力は、その成果が必ずしも評価されるとは限らないのに対し、そのために生じるリスクは明確かつ重大であるため、翻訳者は文脈対応の問題への対処に消極的にならざるを得ないのである。

#### 4.2 文脈対応への配慮を動機づけるために

翻訳者が文脈を重視した訳出に積極的になれるかどうかの背景には、リスクと報酬のバランスが存在する。先に指摘したように、翻訳メモリツールが使用されている場合、ツールの性質や機能上の制限から翻訳者のリスクが大きくなる可能性がある。一方、報酬につながる作業評価においては、文脈対応に配慮した努力は評価されにくい傾向があることも述べた。

つまり、訳出における文脈対応への配慮を動機づけるためには、そのために翻訳者が直面するリスクを少なくし、かつ、報酬につながる明確な作業評価が行われるようにする必要がある。現状において実務翻訳の現場に存在する条件のうち、こうした役割を果たしていると思われるものとして、スタイルガイドが挙げられる。ローカリゼーションなど大容量の翻訳案件が一定期間ごとに繰り返されるようなプロジェクトでは、用語リストに加え、文字使いや特定の文脈に応じた

文体などのスタイル上の仕様を示したスタイルガイドが提供されることがある。例えば前述したコロン(:)によって1つの原文が2つ以上の分節に区切られてしまっている問題についても、翻訳依頼主によっては過去の経験などから予めこれを予測し、翻訳者の対処方法をスタイルガイドに指定している場合がある。コロン(:)を含む原文が複数の分節に分割されている場合は分節を結合して日本語として自然な語順になるように訳出する、などのように具体的な方策を予め指定しておけば、翻訳者は分節の制限を超えることに対してリスクを感じずに日本語として自然な訳出を心がけることができる。同時に、分節を結合して訳出した翻訳者の努力に対しても、「スタイルガイドの仕様を守っている」という明確な評価基準が生まれ、作業成果が評価され易くなるとも考えられる。しかし、文脈上の問題は読んでみないとわからないため、そのすべてを事前に予測して仕様を指定することは不可能であり、仮にある程度まで可能であったとしても、あまりにも指定条件が多く細かすぎるスタイルガイドは訳出および編集作業の効率を著しく低下させることにもなりかねないため、対応できる範囲には限りがある。

では翻訳者のリスクを少なくして、文脈対応に積極的になれるようにするにはどうすればよいだろうか。先に指摘したように、翻訳メモリツールからの影響に限って言えば、翻訳者が特定の文脈に合わせた訳出を行ったり、そのために翻訳メモリツールの制限を超えるような介入をすることで、その結果が意図しなかったところで訳文の再利用性の低下や誤訳などの結果を招くという点がリスク要因になっていると考えられる。このリスクを避けるためには、そういう操作が必要になったとき、その意図を人からも機械からも追跡可能な形で残し、訳文の再利用時に必ず人が介入し判断を下せるような手がかりを作っておけばよいのではないか。例えば SDL Trados の場合、各分節の付帯情報としてユーザーが任意に設定できる属性とペナルティの設定を駆使することで、前述の「Yes」「No」の訳文の重複の問題のような場合に、原文と訳文の1対複数のペアに重みをつけて翻訳メモリに登録しておき、自動翻訳など機械による一括処理を行う際に、人による介入(文脈に適切な訳文の選出)が必要なポイントを機械的に検出・追跡できるように配慮することが可能である(内田 2004)。しかし、このような設定を行うこと自体が翻訳メモリに対する大幅な介入となるので、顧客の資産であるという翻訳メモリの性質上、翻訳者主導でこのような操作を行うことは難しく、一般的に実用されることはほとんどない。現状では、翻訳メモリを既定の設定よりも踏み込んだ形で活用することの実用性、可能性が翻訳者側の視点で議論されることはあまりなく、その結果、資産としての翻訳メモリの管理効率という視点に立った機能性ばかりが重要視されるきらいがある。翻訳支援ツールの本来の性質を見直す意味でも、人間の翻訳者にしかできない文脈対応に配慮した自然な訳文の実現のために、人による翻訳ツールへの積極的な介入をいかに効率的・容易にできるかという観点からの研究が進み、翻訳ツールの機能性および翻訳ツールを使用した翻訳技法のあり方が発展していくことを期待している。

## 5. まとめ

本稿は翻訳メモリツールを使用した翻訳の問題点として、(1) 分節単位で原文と向き合い訳出するというインターフェイス上の制約が文脈の認識および再構成の妨げになる、(2) 訳文の再利用効率への配慮から特定の文脈への依存を極力排除しようとする意識が翻訳者に働く

場合がある、(3) 既存訳が優先されるという点を指摘し、特に (3) については、既存訳が文脈対応という点で見ると不自然ではあるものの厳密には誤訳ではないという場合ほど、そのまま容認される確率が高くなる可能性を示した。

訳出作業の問題が解消されるはずの編集作業についても、実際は翻訳メモリツールの影響下で行われているという現状を示し、編集作業環境としての翻訳メモリツールの非効率性の問題として、編集対象のコンテンツが複数箇所に分散している、分節を翻訳単位とする制約が継続している、必要な付帯情報の表示において編集効率が配慮されていないといった点を指摘した。また、訳出作業時と同様に分節という翻訳単位や訳文の再利用性への配慮といった制約が編集作業時においても継続していることも考察した。

訳出および編集作業において、翻訳メモリツールが翻訳者の意識に及ぼす影響という面では、その背景として翻訳者のリスクと報酬のバランスという考え方を示し、ツールの性質や機能上の制限から翻訳者のリスクが大きくなる可能性があること、一方報酬につながる作業評価においては文脈対応に配慮した努力が評価されにくい傾向がみられることを指摘した。

本稿において考察された諸問題に共通しているのは、翻訳メモリツールの効率を守るために、いかに翻訳者がツールの制限に訳出を寄り添わせるかという「ツールありき」の姿勢である。翻訳メモリツールはまだ歴史が浅く、研究データには限りがあるが、翻訳実務の現場ではすでになくてはならない存在であり、翻訳教授法にも今後積極的に導入されていくことが予想される。その中で、翻訳の主体はあくまでも翻訳者であるという視点から、ツールに求められる機能性やツールを使用した翻訳のあり方について研究が進められ、本稿がその一助となることを願う。

.....

**【謝辞】**本稿執筆にあたり、立教大学大学院異文化コミュニケーション研究科の山田優氏からご協力をいただきましたことに深く感謝を申し上げます。

**【著者紹介】**

内田順子 (UCHIDA Naoko) Monterey Institute of International Studies (MIIS: モントレー国際大学)より 1997 年翻訳通訳修士号を取得。American Translators Association (アメリカ翻訳者協会) 公認英日翻訳者。連絡先: [naoko@japaneseit.net](mailto:naoko@japaneseit.net)

**【註】**

1. SDL Trados では、分節を開く(翻訳できる状態にする)と原文セグメントはブルーでハイライトされたボックス、訳文セグメントは黄色(完全一致なし)または緑色(完全一致メモリあり)でハイライトされたボックスで囲まれて表示される。SDL WorldServer などのように、原文と訳文が左右に並べて表示され、カーソル移動のみで分節が翻訳モードに入り、翻訳中の原文および訳文のハイライト表示は起こらないツールも存在するが、翻訳中の分節に連動して用語ベースや翻訳メモリの一一致データが表示される点ほどの翻訳メモリツールでも共通しており、これらの情報にも目を向けながら翻訳することで、ある程度の「文脈からの分離」が発生すると考えられる。

2. 翻訳会社の作業工程あるいは使用するツールによっては、翻訳メモリ自体は納品しない場合もある。しかしこの場合も、依頼主が納品された翻訳ファイルから翻訳メモリを生成できるように、原文と訳文が対になった状態で保存されている、翻訳メモリツール専用のファイル形式で納品することが求められる。このことは、使用されるツールにもよるが、結果的に納品物の作業はすべて翻訳ツールのインターフェイス内で行う必要があることを意味する。
3. 表示される情報項目はカスタマイズ可能であるため、翻訳メモリツールの設定により異なる。

#### 【参考文献】

- Bowker, L. (2002). *Computer-aided translation technology: A practical introduction*. Ottawa: University of Ottawa Press.
- Bowker, L. (2005). Productivity vs. Quality: A pilot study on the impact of translation memory systems. *Localisation Focus*, 4 (1): 13-20.
- Heyn, M. (1998). Translation memories: Insights and prospects. In L. Bowker, M. Cronin, D. Kenny, & J. Pearson (Eds.), *Unity in diversity? Current trends in translation studies*. Manchester: St. Jerome. 123-36.
- LISA (2004a). Building Quality: The LISA QA Model 3.0. *Globalization Insider* 13 (2.2) [Online] [http://www.lisa.org/globalizationinsider/2004/05/mission\\_impossi.html](http://www.lisa.org/globalizationinsider/2004/05/mission_impossi.html) (July 20, 2009)
- LISA (2004b). LISA Best Practice Guide - Quality Assurance: The Client Perspective. [Online] <http://www.lisa.org/Best-Practice-Guides.594.0.html#c1773> (July 20, 2009)
- Pym, A. (2008). On Toury's laws of how translators translate. [Online] [http://www.tinet.org/~apym/on-line/translation/2007\\_toury\\_laws.pdf](http://www.tinet.org/~apym/on-line/translation/2007_toury_laws.pdf) (July 20, 2009)
- Weltz, R. N. (2001). The Proof is in the Reading. *ATA Chronicle* 30 (4): 12-13.
- Sirena, D. (2004). Mission Impossible: Improve Quality, Price and Speed At the Same Time Using SAE J2450 to Do the Impossible. *Globalization Insider*, 13 (2.2) [Online] [http://www.lisa.org/globalizationinsider/2004/05/mission\\_impossi.html](http://www.lisa.org/globalizationinsider/2004/05/mission_impossi.html) (July 20, 2009)
- 内田順子(2004)「Trados における重複する翻訳メモリの管理」[Online] <http://www.japaneseit.net/files/ATALecture.pdf> (July 27, 2009)
- 機械翻訳課題調査委員会(2009)「2008 年度活動報告」『AAMT アジア太平洋機械翻訳協会第 19 回通常総会資料』アジア太平洋機械翻訳協会: 22.
- 玉置裕子(2004)「体系的翻訳論への一試論 —情報構造および関連性理論を中心に—」『通訳研究』日本通訳学会 第 4 号: 157-169.
- 山田優(2008)「翻訳メモリ使用時の既存訳が新規訳に及ぼす影響 —干渉と翻訳の普遍的特性の観点から—」『通訳翻訳研究』日本通訳翻訳学会 第 8 号: 191-207.
- 山田優(2009)「翻訳技術の翻訳プロセスへの影響と可能性」『翻訳研究への招待』日本通訳翻訳学会 翻訳研究分科会 第 3 号:133-144.
- 脇田早紀子(2009)「翻訳品質の定量評価 —ツールによる納品検査の成功／挫折と現状」『AAMT ジャーナル』アジア太平洋機械翻訳協会 第 45 号: 4-8.

