

「ムギはかわいい」の非対称性と非総記性¹

Junichi Iida

以下の 1 では「ムギならばかわいい」という読みと「かわいいならばムギである」という読みが同時に成立する。これに対し 2 では「ムギならばかわいい」は読み取れるが、「かわいいならばムギである」は読み取れない。このような個別的な事実は、語彙項目ごとあるいは構文ごとに直接規定されるのではなく、一般規則から機械的に導出されることが望ましい。理論が仮定する形式体系が個別の文に関する素朴な事実を正しく予測するかどうか、またその導出にどのような法則が関係し、どのような一般化が得られるかを確認することは、対象言語についての知識を蓄積する方法の一つである。

1. ムギしかかわいくない。 (Semantics, pp. abstainer-bunnies)
2. ムギはかわいい。

筆者は Semantics (Iida 2026) で 1 について次の解釈を与えた。

3.
$$\begin{cases} \text{Mugi is cute} \\ \text{Mugi' is not cute} \\ \text{Mugi'' is not cute} \end{cases} \quad (\text{p. apple})$$

Mugi はムギを指し、Mugi' および Mugi'' はムギに対比されたり累加されたりしうる文脈上の存在を指す。Mugi に対して is cute あるいは is not cute という評価が与えられるとき、その文脈には同様の評価を付与されうる対象として Mugi' と Mugi'' が存在する。Mugi, Mugi', Mugi'' の三者からなる三つ組 [Mugi, Mugi', Mugi''] を Mugi の context という。これは言語外の知識におけるその照応物に対応し、Semantics (p. apple) では [ムギ, ビビ, ルナ] を Mugi の context として考えた。Mugi' および Mugi'' のプライム記号は context のうちの何番目の項目をそのラベルが指すかを示す符号であると言える。プライム記号付きのラベルが何を指すかは暗黙であり、言語がそれを知っている必要はない。言語は Mugi' というラベルについて「Mugi の context において Mugi でないなんらかの対象が照応されている」ことを知っており、また Mugi' と Mugi'' が同時に現れた場合について「Mugi の context において Mugi 以外のすべての対象が照

¹ An official document of Kyouro/教育ローマ字; Version 7.2.1, originally released as Version 7.2.0 in August 2026, updated to August 2026.

応されている」ことを知っている。Mugi'' が単独で使われることはない (Convention 1.2, p. ditto)。3 では Mugi について is cute という評価が与えられ、Mugi' と Mugi'' の両方について is not cute という評価が割り当てられているため 3 全体ではおよそ「ムギがかわいい。ムギ以外の全員がかわいくない」という意味に相当する。

3 は非形式的な記述だが、相当する形式的な意味表示は次のように導出される。

$$\begin{array}{l}
 4. \\
 \text{mugi} \quad \text{shika} \quad \text{kawaiku} \quad \text{nai} \\
 \text{Mugi} \quad \text{it} \quad X: \frac{X', X''}{X} \quad \text{cute} \quad \text{is} \quad X: \frac{\neg X}{X} \quad . \\
 \begin{array}{c}
 \text{itMugi} \qquad \qquad \text{iscute} \\
 \frac{\text{itMugi}', \text{itMugi}''}{\text{itMugi}} \qquad \frac{\neg \text{iscute}}{\text{iscute}} \\
 \frac{\text{itMugi}', \text{itMugi}'' \cdot \neg \text{iscute}}{\text{itMugi} \cdot \text{iscute}} \\
 \left[\begin{array}{c} \text{itMugi}' \cdot \neg \text{iscute} \\ \text{itMugi}'' \cdot \neg \text{iscute} \\ \text{itMugi} \cdot \text{iscute} \end{array} \right] . \\
 \left[\begin{array}{c} - \text{itMugi}' \cdot \text{iscute} \\ - \text{itMugi}'' \cdot \text{iscute} \\ + \text{itMugi} \cdot \text{iscute} \end{array} \right] . \quad \text{———— (4a)}
 \end{array}
 \end{array}$$

4a には 3 に相当する真理条件の情報が含まれる。この構造物は proposal といい、entailment と呼ばれる内部構造を含む。- itMugi' · is cute は entailment の一つである。Entailment はプラスもしくはマイナスの符号と proposition と呼ばれる構造からなる。ドットで区切られた itMugi や iscute は segment と呼ばれ、proposition の構成要素である。Entailment は proposition について真 (+) か偽 (-) かをあらかじめ述べるものである。節がそれぞれこのような proposal を持ち、entailment の真偽は節に対して割り当てられる event と呼ばれる構造を元とする集合について評価される。Event は segment を元とする集合であり、割り当てられた event の集合の中に、その節に属する proposition に含まれる segment をすべて含む event が少なくとも一つ存在すれば、その proposition は真 (+) であり、そうでなければ偽 (-) である。Entailment は proposition の真偽を正しく言い当てていれば真、そうでなければ偽となる。(この一連の仕組みは Chapter Two で述べた。) 4a は、itMugi' と iscute を両方含む event が割り当てられた集合の中に存在せず、同様に itMugi'' と iscute を両方含む event が存在せず、itMugi と iscute を両

方含む event が存在すると述べていることになる。このようにして節の真理条件は集合同士の包含関係の条件として定義される。

4a のすべての entailment が真であるとき、Mugi の context において cute なのは Mugi だけであるから、冒頭で指摘したように、cute ならば Mugi であり ($\text{Mugi} \leftarrow \text{cute}$)、Mugi ならば cute である ($\text{Mugi} \rightarrow \text{cute}$) ということが ₁ の意味において成り立つ。括弧内はそれぞれの関係を簡単に記号化したものだが、節が両者を同時に主張することを次のように表すことにする。

5. $\text{Mugi} \leftrightarrow \text{cute}$

5 はムギについてかわいいと述べるあらゆる節において成り立つわけではない。すでにみたように、₂ においては $\text{Mugi} \rightarrow \text{cute}$ が意味され、 $\text{Mugi} \leftarrow \text{cute}$ が意味されない。ここに Mugi と cute の間の非対称性がある。₁ が ₅ を満たすことを導いた同じ規則体系が ₂ が ₅ を満たさないことを導出することを確認することがここでの課題となる。

₂ の意味表示は次のように導出される。(p. leeway の *kōhī wa oishii* などが同型である。)

$$\begin{array}{l}
 6. \\
 \text{mugi} \quad \text{wa} \quad \text{kawaii} \\
 \text{mugi} \text{ it } X: \frac{X}{X'} \quad \text{cute} \text{ is } X: \frac{X}{\neg X} \quad . \\
 \text{itMugi} \quad \text{iscute} \\
 \frac{\text{itMugi}}{\text{itMugi}'} \quad \frac{\text{iscute}}{\neg \text{iscute}} \\
 \frac{\text{itMugi} \cdot \text{iscute}}{\text{itMugi}' \cdot \neg \text{iscute}} \\
 \left[\begin{array}{l} \text{itMugi} \cdot \text{iscute} \\ \text{itMugi}' \cdot \neg \text{iscute} \end{array} \right] . \\
 \left| \begin{array}{l} + \text{itMugi} \cdot \text{iscute} \\ - \text{itMugi}' \cdot \text{iscute} \end{array} \right| . \quad \text{———— (6a)}
 \end{array}$$

₂ が ₅ を満たさないことは 6a から直接導かれる。6a は Mugi について cute であると述べ、Mugi' について cute でないと述べているものの、Mugi'' については何も述べていない。そのため、6a のすべての entailment が真であるとき、割り当てられた event の集合の中に itMugi'' と iscute を含む event がある場合が排除されない。したがって、節の意味としては $\text{Mugi} \leftarrow \text{cute}$ が成立せず、₅ も成り立たないということになる。₂ が ₅ を満たさないことが導出されることが確認された。

一般に、 X の context の要素について Y であるか否かを述べる節について、 $X \leftrightarrow Y$ が成り立つことは、 X がプラスの指定を受けた entailment に配置され、 X' および X'' がマイナスの指定を受けた entailment に配置されることと同値である。これを満たすためには X の context の要素がすべて参照されることが必要条件となる。このような参照を exhaustive reference と呼び、そうでない参照を non-exhaustive reference と呼ぶ (p. ditto)。

5 を 1 が満たし 2 が満たさない理由は、音韻形式 *shika* と *wa* にそれぞれ語彙的に対応づけられたノード $X: \frac{X', X''}{X}$ と $X: \frac{X}{X'}$ の構造の違いに帰着される²。表記に関する詳細は Chapter One に譲るが、これらのノードは context $[X, X', X'']$ の要素のうちどれが参照されるかをその表記で直接示しており、*wa* において三要素のすべてが参照されるわけではないことは構造表記内に X'' が現れないことから直接読み取れる。同様に、*shika* においては三要素が参照されることがわかる。1 においては、さらにこれらの三要素が 4a において先述の $X \leftrightarrow Y$ の成立条件を満たす形で配置される。1 が 5 を満たすのはこのためである。一方、2 においては *Mugi''* が参照されないため、 $X \leftrightarrow Y$ の必要条件を満たさない。2 が 5 を満たさないのは、*wa* による context の参照が non-exhaustive (非総記的) だからであると言える。

本稿では、「ムギしかかわいくない」と「ムギはかわいい」のうち、前者についてのみ $\text{Mugi} \leftrightarrow \text{cute}$ が成り立つことが Semantics の体系内で必然的に導出されることを示し、 $\text{Mugi} \leftrightarrow \text{cute}$ を一般化した $X \leftrightarrow Y$ の必要十分条件を述べ、これを「しか」が満たしうが「は」が満たしえないことを指摘した。非総記性（言及の対象にならない要素があること）と一意性の保証（ X が、そして X のみが Y であると言えること）が互いを排除することは直感的に自明だが、文法におけるその帰結は形式化してみなければ意識しにくい。参照には総記的 (exhaustive) なものと非総記的なものがあり、後者が一意性の保証を排除するという事実には、他にも無視できない文法上の帰結があるものと期待される。なにしろ、文法は参照と論理の体系なのだから。

References

Iida, J. 2026 Semantics of Tokyo Japanese, ver. 6.5.6. <https://nihongotopics.github.io/Kyouro/Semantics%20of%20Tokyo%20Japanese.pdf>.

² これらの表記はいずれも Semantics ですすでに与えられている。実例は p. leeway などを参照されたい。