
Transcriptional Implicature: Using a Transcript to Reason about an Exemplar

Claus Huitfeldt
Claus.Huitfeldt@uib.no
University of Bergen, Norway

C. M. Sperberg-McQueen
cmsmcq@acm.org
Black Mesa Technologies LLC
United States of America

What do transcripts tell us about their exemplars, and how? A transcript T is commonly said (inter alia) to reproduce as far as possible the letters and words of its exemplar E. The tokens of T typically reinstantiate the letter and word types of E; T may also contain commentary on E.

Transcribers vary in their practice: some record page and line breaks in E, silently omit deletions, expand abbreviations, and correct spelling; others don't. Often a statement of practice documents such details, but some things apparently go without saying. We conjecture that what goes without saying will be the most common assumptions and habits in a community of practice. We have proposed the term "transcriptional implicature" to denote the inferences licensed by a transcript although not explicitly justified in the statement of practice [SMMH2014].

Practice (and implicature) vary across communities, but we think that some silent assumptions are shared by almost every community of practice. We believe that we can identify a common core of such assumptions, and that individual practices can be characterized by listing their deviations from this core, which we call the "default rules" of transcriptional implicature.

Like all implicatures, the rules of transcriptional implicature are defeasible: they are taken to hold in the absence of evidence to the contrary. [SMMH 2014] proposes these rules:

- Reciprocity: There is a one-to-one relation between normal tokens in an exemplar E and normal tokens in its transcript T.
- Purity: Every normal token in T transcribes something in E.
- Completeness: Every normal token in E is transcribed by something in T.
- Type similarity: corresponding tokens in E and T have the same type, or non-identical but similar types.

Given these rules, the transcription practice of any project can be described by defining what counts for that project as a special (not "normal") token and what type pairs count as similar. This paper puts these general principles to an empirical test by applying them to a concrete example: the transcript, in the Northwestern-Newberry Edition, of Hermann Melville's notes on the end-papers of a volume of Shakespeare [Melville, p 967-970]

The "Symbols used" are (numbers ours):

1. [...] revision or insertion enclosed in square brackets was made later than initial inscription of leaf
2. <...> letters or words enclosed in diamond brackets were canceled by lining out
3. <...>word letter(s) or word(s) written over are enclosed in diamond brackets closed up to the following word or letter that was superimposed
4. ?word prefixed question mark indicates conjectural reading 5 xxxx undeciphered letters (number of x's approximates numbers of letters involved)
5. all words in roman are Melville's
6. all words in italics outside brackets are words Melville underlined
7. all words in italics inside brackets are editorial

If transcriptional implicature is to play the role we ascribe to it, it must be possible to recast these rules in terms of normal tokens, special tokens, and types instantiated by tokens:

- Square brackets, diamond brackets, question marks prefixed to words, and sequences of the form xxxx in T are special tokens; they transcribe no material in E.
- Italic material inside brackets in T is special.
- A word written without underlining and the same word underlined have distinct types in E; the same holds for italic and roman

material in T. Words underlined in T and the same words italicized in E are type-similar.

- Later additions to E have distinctive types; that is, we can distinguish them. A sequence of words inserted in E and the same sequence enclosed in square brackets in T are type-similar.
- Overlined words, overwritten words, and words neither overwritten nor overlined instantiate different types. Overlined or overwritten word-types are instantiated in T by enclosing the words in diamond brackets.
- "Undeciphered letter sequences of length N" are distinct types for distinct N, instantiated in E by undeciphered characters and in T by sequences of the form xxxx. The individual x tokens are special: they do not transcribe tokens of E.
- A word prefixed with a question mark and the same word without it instantiate different types in T. Such a word token, and its constituent character tokens, are type-identical with their exemplars in E if the editors' conjectural reading is correct. We describe such tokens in T as "probably type-identical" to their exemplars.

Both generic and project-specific rules can be formalized using first-order logic. The generic rules include the rules of transcriptional implicature:

```
(∀t t:tokens(T))(normal-transcript-token(t) ⊃
  (∃e e:tokens(E))(e = exemplar(t)))
(∀e e:tokens(E))(normal-exemplar-token(e) ⊃
  (∃t t:tokens(T))(t = transcript(e)))
(∀e e:tokens(E), t:tokens(T))(t = transcript(e) ⊃
  (type(e) = type(t) ∨ type-similar(e, t)))
```

Reciprocity has no formalization here; it's a property of the functions `exemplar()` and `transcript()`. Other generic rules constrain the classes of normal and special tokens:

```
(∀t t:tokens(T))(normal-transcript-token(t) |
  special-transcript-token(t))
(∀e e:tokens(E))(normal-exemplar-token(e) |
  special-exemplar-token(e))
```

The Melville-specific rules identify special tokens in T:

```
(∀t t:tokens(T))(special-transcript-token(t) ⊃
  (square-bracket(t)
  ∨ angle-bracket(t)
  ∨ prefixed-question-mark(t)
  ∨ xxx-sequence(t)
  ∨ (italic(t) & within-square-brackets(t))
  ∨ (italic(t) & within-angle-brackets(t))
  ∨ prefixed-question-mark(t)
  ∨ page-furniture-token(t)))
```

The predicate `page-furniture-token` identifies material like running heads and page numbers in T. Project-specific type similarity rules cover conjectural readings:

```
(∀t t:tokens(T))(∀e e:tokens(E))
  (probably-identical(type(t), type(e)) ⊃
  type-similar(e, t))
```

The Melville transcription policy defines no special exemplar tokens:

```
¬(∃t t:tokens(E))(special-exemplar-token(t))
```

We hypothesize that it is rules like those above that enable readers of a transcript to draw conclusions about an exemplar they have never seen. Such reasoning can be formalized with the aid of a full formal representation of T and its mapping of tokens to types. In the case of Melville's notes, such a formalization will involve several thousand tokens in T and E and their types. The formal representation will not be given in full here, but it will include formulae like the following, in which "d", "p524", "L1", etc. are logical constants naming tokens:

```
document(d)
page(p524)
page(p525)
document_pages(d, (p524, p525)).
line_token(L1)
line_word_tokens(L1, (L1w1, L1w2, L1w3, L1w4, L1w5, L1w6, L1w7))
word_token(L1w1)
word_token(L1w2)
...
word_token_type(L1w1, "A")
word_token_type(L1w2, "seaman")
...
word_token_type(L1w7, "Tales.")
line_token(L2)
...
line_token(L40)
line_type(LT1)
line_type(LT2)
...
line_type(LT40)
page_lines(p524, (L1, L2, L3, ... L30))
page_lines(p525, (L31, L32, L33, ... L40))
```

Sequences are here represented using comma-separated lists of items, enclosed in parentheses. As an example: We can infer from the transcript that the first line of the first page transcribed reads "A seaman figures in the Canterbury Tales." This will

surprise no one who can read and understands what a transcript is. The challenge here, however, is to establish the result using nothing but the normal rules of logical inference, without hand-waving.

Let us focus on the line token L1. We draw our initial inferences from the formal representation of T. As is usual for composite types, the type of the line is determined by the types of the words which make up the line. So the type of L1 is the sequence of words (or characters) "A seaman figures in the Canterbury Tales."

```
composite_type_string(LT1, "A seaman figures in the Canterbury Tales.")
```

Since L1 is not a square bracket, diamond bracket, etc., it can be inferred that L1 is a normal not a special token. From the rule of purity it then follows that there is some token e in E which L1 transcribes. This fact, together with the rule of type-similarity, tells us that e is a token which like L1 instantiates the composite type "A seaman figures in the Canterbury Tales."

A second example: A student examining a reproduction of the original of Melville's notes might ask "Are the dots to the left of the first line significant, or just discolorations of the paper?" We take this question to mean "Do the dots instantiate a type?" Let us assume that they do. If they instantiate a type, then they are tokens in E; if they are tokens in E, then either they are normal or special. They cannot be special tokens: there are none in this transcription practice. If they are normal tokens, then there exists a token in T which instantiates the same type. But there are no such tokens in T. If there is no token in T which transcribes the dots at the upper left of the page, then they are not (in the transcribers' reading of E) normal tokens. If they are neither normal tokens nor special tokens, they are not tokens at all. So: they are merely marks, not tokens, and they instantiate no type.

We have postulated transcriptional implicature as a way of bridging the gap between the explicit statements in descriptions of transcription practice and the inferences actually licensed; the empirical test reported suggests that the concept does provide the required basis for the intended inferences. If formal representations of the information content of a transcript can be generated automatically (e.g. from a TEI-encoded transcript), then we will be slightly closer to being able to describe formally the semantics of transcription-oriented markup languages like MECS-WIT or TEI when used to transcribe pre-existing exemplars.

Bibliography

- Melville, H** (1998) *Moby Dick, or, the Whale*. Ed. Harrison Hayford, Hershel Parker, and G. Thomas Tanselle. Vol. 6 of *The Writings of Herman Melville, The Northwestern-Newberry Edition* 1988, rpt. 1994, 1997.
- Huitfeldt, C.** (1993). MECS-WIT - A Registration Standard for the Wittgenstein Archives at the University of Bergen, 1993 approx. 250 pages. Available electronically only, now at <http://folk.uib.no/fafch/oldstuff/mecswit.html>
- Sperberg-McQueen, C. M., Marcoux, Y., and Huitfeldt, C.,** (2014). "Transcriptional Implicature: A Contribution to Markup Semantics." Paper at DH 2014, Lausanne. Abstract on the Web at <http://dharchive.org/paper/DH2014/Paper-61.xml>
- TEI Consortium,** eds (n.d.) *TEI P5: Guidelines for Electronic Text Encoding and Interchange*. [Version 3.1.0. Last updated on 15th December 2016, revision d3f5e70]. TEI Consortium. <http://www.tei-c.org/Guidelines/P5/>.