PyCantonese: Developing computational tools for Cantonese linguistics

Jackson L. Lee, Litong Chen, Tsz-Him Tsui
University of Chicago and The Ohio State University

The 3rd Workshop on Innovations in Cantonese Linguistics
The Ohio State University
March 12, 2016

What is missing in Cantonese linguistics?
Name subfields with lots of work on Cantonese!

phonetics, phonology, morphology, syntax, semantics, pragmatics, sociolinguistics, historical linguistics, discourse and conversation analysis...

How about...

Computational linguistics?

We are concerned with the strongly empirical and data-driven kind of computational linguistics.

Why computational linguistics? Why data?
Reproducible research

• Verifiable claims in linguistic research

Modeling learnability

• How does grammar come from data?

The socio-political status of Cantonese (?)

• Preserving data → Protecting and promoting the language

Apparent lack of computational linguistics for Cantonese
': Lack of data?

We do have data! (And we need more...)

Several Cantonese corpora
Adult Cantonese:

• The Hong Kong Cantonese Adult Language Corpus (Leung and Law 2001; Leung et al. 2004; Fung and Law 2013)
• Cantonese Radio Corpus (Francis and Matthews 2005, 2006)
• PolyU Corpus of Spoken Chinese (Yap et al. 2014)
• Hong Kong Cantonese Corpus (Luke and Wong 2015)
Child developmental data:

- Hong Kong Cantonese Child Language Corpus (Lee and Wong 1998)
- The Hong Kong Bilingual Child Language Corpus (Yip and Matthews 2007)

Non-contemporary Cantonese:

- Early Cantonese Tagged Database (Yiu 2012)
- A Linguistic Corpus of Mid-20th Century Hong Kong Cantonese (Chin 2013)

So, what is missing?

Comparing some Hong Kong Cantonese corpora
Both standard and non-standard data formats have been used.
Using multiple corpora in research?
It’s hard!

∵ Individual corpora are usually compiled for specific purposes
⇒ Different foci in annotations and formatting

Some recent work that could have benefited from more data:

• Chen (2015): phonological variation of keoi5 ‘s/he’ in HKCAC
• Tsui (2014): functional load of Cantonese tones in HKCanCor

PyCantonese – General goals

Data format
PyCantonese adopts the CHILDES CHAT format (MacWhinney 2000).

• Rich annotations for conversational data
• Well documented and supported
• PyCantonese piggybacks on PyLangAcq (Lee et al. 2016) for handling the CHAT format.

(How about non-conversational data?)

PyCantonese – Background
PyCantonese is a growing toolkit for computational work in Cantonese linguistics.

• It is a Python library – why Python?
  – a general-purpose programming language
  – the lingua franca for computational linguistics and natural language processing
• Similar data structures as in NLTK (Bird et al. 2009)
• A free and open-source tool
• Full documentation (with installation instructions): http://pycantonese.org/
Basic functionality
PyCantonese comes with built-in corpus data. Currently, KK Luke’s HKCanCor is included.

For some given corpus data, we can ask about its basic information...

Let’s begin...

```python
>>> import pycantonese as pc
>>> corpus = pc.hkcancor()
>>> corpus.number_of_files()
58
>>> corpus.number_of_utterances()
15938
```

Accessing corpus data
words()

```python
>>> all_words = corpus.words()
>>> len(all_words)
149781
>>> all_words[:10]
['喂', '遲', 'o的', '去', '唔', '去', '旅行', '啊', '?', '你']
```

characters()

```python
>>> all_characters = corpus.characters()
>>> len(all_characters)
186888
>>> all_words[:10]
['喂', '遲', 'o的', '去', '唔', '去', '旅行', '啊', '?']
```

Word-level annotations
tagged_words()

```python
a tagged word =
(word, part-of-speech tag, Jyutping, grammatical relations)

>>> all_tagged_words = corpus.tagged_words()
>>> all_tagged_words[:4]
[('喂', 'E', 'wai3', '), ('遲', 'A', 'ci4', '), ('o的', 'U', 'di1', '), ('去', 'V', 'heoi3', ')]
```

(More on grammatical relations in a minute!)

Other methods: http://pycantonese.org/reader.html
— utterance-level structures, word frequency info, etc.
Parsing Jyutping

```python
parse_jyutping()
```

Jyutping → (onset, nucleus, coda, tone)

```python
>>> import pycantonese as pc
>>> pc.parse_jyutping('hou2')
[('h', 'o', 'u', '2')]
```

```python
>>> pc.parse_jyutping('hoeng1gong2')
[('h', 'oe', 'ng', '1'), ('g', 'o', 'ng', '2')]
```

Search queries

Possible search queries depend heavily on what is encoded and annotated in the corpus data:

**Jyutping elements? Part-of-speech tags? Characters?**
A combination of any of these?

Additional features:

- Search by a word/sentence range
- Search by a regular expression

Details — [http://pycantonese.org/searches.html](http://pycantonese.org/searches.html)

Example: jau5 ‘have’, C. Lam (2016a) 1 hour ago
Example: aa is the only onsetless syllable with all 6 tones in HKCanCor, cf. Z. Lam (2016b) 2 hours ago

Ongoing work

- Corpus reformating (currently the HKCAC dataset)
- Devising tools for filling in the gaps in formatting and annotations across corpora

Anticipated functionality

- Jyutping ↔ characters (issues: homophony and homography)
- word segmentation (a perennial problem for CJK languages)
- part-of-speech tagging (depending on tagset etc)

We’d need these for preparing a usable corpus dataset based on, say, the novel 男人唔可以窮 from the HK Golden Forum!

More on the to-do list

- Forced alignment (cf. Peters and Tse (2016) 30 min ago)
- Dependency and grammatical relations
Moving Cantonese linguistics forward

- We all need one another.
- PyCantonese opens the door for
  
  shared and open-access resources.
- Call for arms!
  
  PyCantonese is a collaborative project.
- Questions, comments, bug reports, feature requests etc
  are more than welcome.

References


Yap, Foong Ha, Ying Yang and Tak-Sum Wong. 2014. On the development of sentence final particles (and utterance tags) in Chinese. In Kate Beeching and Ulrich Detges (eds.), *Discourse functions at the left and right periphery*, 179-220. Leiden: Koninklijke Brill NV.
