


TEACHING TEXT-FABRIC TO THEOLOGICAL STUDENTS

Christian Canu Højgaard

cch@dbi.edu

Digital Hermeneutics and the Reception of the Hebrew Bible

 jupyter Basic Python and Text-Fabric Last Checkpoint: 18.11.2018 (unsaved changes)

 Logout

File Edit View Insert Cell Kernel Widgets Help
[Icons] [Run] [Markdown]

Basic Python and Text-Fabric

By Christian Højgaard Jensen (chj@dbi.edu)

Adapted from Martijn Naaijer (https://github.com/MartijnNaaijer/Shebanq_Course_Files/blob/master/Introduction_to_text_fabric)

Welcome to this course on text-fabric. This course will teach you how to extract data from the Eep Talstra Centre for Bible and using the Python package text-fabric. You do not need to have any Python knowledge to do this course, because you will learn the same time.

Text-fabric was developed by Dirk Roorda and Wido van Peursen as part of the SHEBANQ-project. In this project the website was developed. On this website you can inspect the text of the Hebrew Bible and you can make queries on this text with the feature the Mini Query Language (MQL). Text-fabric serves as a research tool to make datasets that can be analyzed further. The website is available at the website of [Data Archiving and Networked Services \(DANS\)](#), from where it can be downloaded.

In this course you will first learn some basic Python, and then move to text-fabric as soon as possible.

About Python

We start with some basic things you need to know about the Python language. It includes basic data types, data structures and data flow control. When you

- Autumn 2018
- 7 students
- Text-Fabric browser
- Jupyter Notebook tutorials
- Canvas discussion threads

LAF-
workshop
2016



Update Anaconda:

- Anaconda: conda update conda

Update python:

- Anaconda: conda update python

Check ipython version:

- Anaconda: python -version

Install Jupyter:

- pip3 install jupyter

Install LAF-fabric:

- Anaconda: pip install laf-fabric

Open browser:|

- Anaconda Prompt: jupyter notebook

From LAF-Fabric....

```
In [ ]: from laf.fabric import LafFabric
        from etcbc.preprocess import prepare
        from etcbc.lib import Transcription
        fabric = LafFabric(verbose='DETAIL')
```

```
In [2]: source='etcbc'
        version='4b'

        API=fabric.load(source+version, 'lexicon', 'workshop', {
            "xmlids": {"node": False, "edge": False},
            "features": (''
                otype
                lex g_word g_cons g_word_utf8
                sp pdp nametype ls gloss lang
                chapter verse
                '' , 'mother'),
            "prepare": prepare,
            "primary": False,
        }, verbose='DETAIL')
        exec(fabric.localnames.format(var='fabric'))
```

```
0.00s LOADING API: please wait ...
0.28s DETAIL: COMPILING m: UP TO DATE
0.32s USING main DATA COMPILED AT: 2015-11-02T15-08-56
0.32s DETAIL: COMPILING a: UP TO DATE
0.34s USING annox DATA COMPILED AT: 2016-01-27T19-01-17
```

to Text-Fabric

```
from tf.app import use
A = use('bhsa', hoist=globals())
```

Pedagogy

Syllabus

- Theory vs. practice
- Final exam vs. progressive tasks
- Strict curriculum vs. free exploration
- Linguistics vs. exegesis

Digital resources

- Distribution
- Compatibility
- User guides
- Documentation

Profiling a theological student



Computer skills

Linguistic knowledge

Statistical knowledge

RECOMMENDATIONS FOR FUTURE DEVELOPMENTS

1. Feature documentation



BHSA



GitHub
13 Stars · 9 Forks

BHSA

Home

Cantillation and accents

Colofon

Text-Fabric versus SHEBANQ

Updates

References

Words and morphemes

Features ^

0 home

book

book@ll

chapter

code

det

dist

dist_unit

distributional_parent

domain

0 home

This is *continuous* version **c**.

To be updated

Viewable in SH

the weekly upd

Introductio

This is the key t

We organize the

- [grid](#)
- [sectional](#)
- [word](#)
- [lexeme](#)



Table of contents

Introduction

Grid features

Sectional features

Clause (atom)

For *clause*-like objects the values provide a classification on the basis of the constituents of the clause. This feature is also known as *clause (atom) type*.

NOTE

I prefer a more informative definition.

Explain why is this a useful feature? Examples needed.

2. Query examples

#!

Text Words Queries Notes **i**

Log In

01

JHWH is speaking

Oliver Glanz ETCBC Test Project

In this query we are searching for JHWH in subject position and speaking (predication)

> shared ✓ 2014-09-29 18:29:34

> **c** never executed

▼ 2017 **SQL** **i**

```
select all objects
where
[clause
  [UnorderedGroup
    [phrase function = Pred
```

status results up to date

bhsa **i** **c** **2017** 4b 4 **i** English **Oliver Glanz: JHWH is speaking** 1 → **hebrew** **phonetic** **text** Notes (1/3)

1521 words in 749 verses on 38 pages

Genesis 2: 18 **ויאמר יהוה** אלהים לא טוב היות האדם לבדו אעשה-לו עזר כנגדו:

Genesis 3: 13 **ויאמר יהוה** אלהים לאשה מה-נאת עשית ותאמר האשה הנחש השיאני ואכל:

Genesis 3: 14 **ויאמר יהוה** אלהים אל-הנחש כי עשית זאת ארור אתה מכל-הבהמה ומכל חית השדה על-נחנקך תלך ועפר תאכל כל-ימי חייך:

Genesis 3: 22 **ויאמר יהוה** אלהים הן האדם היה כאתר מפניו לדעת טוב ורע ועתהו פן-ישלח ידו ולקח גם מעץ החיים ואכל וחי לעלם:

Genesis 4: 6 **ויאמר יהוה** אל-גין למה תרה לך ולמה נפלו פניך:

Genesis 4: 9 **ויאמר יהוה** אל-מיז אי הבל אחיה ויאמר לא ידעתי השמר אחי אנכי:

Cite

2. Query examples

The screenshot displays the TF-browser interface with the following components:

- Author:** Christian Canu Højgaard
- Title:** Delayed subjects in the Hebrew Bible
- Description:** This query identifies all clauses with a delayed subject, that is, a subject following a predicate, an object and an object phrase, in this particular order.
- verse pad:** Exodus 7:17, Zechariah 4:10, Exodus 32:24
- node pad:** 100, 101, 102
- search pad:**

```
clause
  phrase function=Pred
  < phrase function=Objc
  < phrase
  function=Adju|Loca|Time
  < phrase function=Subj
```
- Control Panel:** expand all verses (checked), expand all nodes (unchecked), expand all results (checked), 2 current position, 20 results per page, Total 20, and a list of results (1-9) with '2' highlighted.
- Results:** A hierarchical tree view showing a 'phrase Time NP' containing Hebrew text and grammatical tags like 'subs four', 'conj and', 'subs day', and 'subs night'. Below it, a 'sentence 6|443' contains a 'clause wqt=0' with a 'phrase Pred VP' (verb wipe, gal, perf) and a 'phrase Conj CP' (conj and).

TF-browser

Query

Export

?

3. Early exposure

Word

Phrase

- Separate lines
- Show border
- Type
- Function
- Determination
- Relation

Clause

Sentence

Clear grammar

Genesis 1

אלהים	ברא	בראשית ¹
הארץ	?	את השמים ואת הארץ:
חשך	?	תהו ובהו
היתה		
רוח אלהים	?	על פני תהום
יאמר	?	על פני המים:
מרחפת		
יתהי	?	אור
יתהי		
אלהים		

4. Tutorials

Get started:

- Launch text-fabric
- Navigate between grammatical layers (word, phrase, clause, etc.)
- Extract data
- Store data in dictionaries/dataframes
- Export data

Not research notebooks
Not database explorations

4. Tutorials

Get inspired:

- Concrete queries for addressing exegetical questions
- Extracting valence patterns
- Discovering simple syntactic patterns

Start analyzing

- Basic distributional statistics
- Basic clustering analyses

5. Knowledge base

Documentation

**Current research -
contact**



**Query examples
and storage**

**Q & A's –
Stackoverflow-
like?**

Tutorials and guides

5. Knowledge base – but how?

The screenshot shows a Stack Overflow page for the question "How to use a Python dictionary?". The page includes a navigation bar with the Stack Overflow logo, links for Products, Customers, and Use cases, a search bar, and buttons for Log in and Sign up. A dark banner at the top of the content area reads "Search less. Build more. Use Stack Overflow for Teams at work to share knowledge with your colleagues. Free 30 day trial. Start your trial." The left sidebar contains navigation links for Home, PUBLIC, Stack Overflow, Tags, Users, Jobs, TEAMS, and What's this? with a Free 30 Day Trial badge. The question text is "I am finding it difficult to iterate through a dictionary in python." and "I have already finished learning via CodeAcademy and solo learn but still find it tough to go through a dictionary." The question was asked 2 years, 7 months ago, is active, and has been viewed 4k times. It has a "python" tag and 7 answers. The user "Roland" edited the question on Jul 13 '17 at 7:14, and "niksy" asked it on Jul 13 '17 at 5:44. A grey callout box on the right contains two bullet points: "Slack for networking and exchanging ideas (closed forum)" and "Another platform for discussing concrete solutions to concrete TF-problems (open access)". At the bottom, a dark banner states: "By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#)."

- **Slack** for networking and exchanging ideas (closed forum)
- **Another platform** for discussing concrete solutions to concrete TF-problems (open access)

5. Knowledge base – but how?

The screenshot shows the Text-Fabric website. At the top left is the DANS logo with the text "Data Archiving and Networked Services". Next to it is the Text-Fabric logo, a circular emblem with a red hash symbol and the letters "TF" in black. To the right of the logo is the text "Text-Fabric". Further right is a search bar with a magnifying glass icon and the word "Search". On the far right is a GitHub repository badge for "annotation/text-fabric" with "40 Stars · 10 Forks".

Below the header is a horizontal navigation menu with buttons for "Home", "About", "Use", "Api", "Writing", "Model", "Create", "Server", "Code", and "Implementation".

On the left side of the main content area, there is a "Text-Fabric Home" link. To its right is a large circular image of the Text-Fabric logo. Below the logo is the text "Text-Fabric" and "About".

Below the "About" section, there is a paragraph of text: "Before diving head-on into Text-Fabric, you might want to read a bit more about what it is and where it came from. And after it, if you want to cite it, use this DOI: [10.5281/zenodo.592193](https://doi.org/10.5281/zenodo.592193)." Below this is another paragraph: "On the other hand, if you do want to dive in, here are the [docstrings](#)."

At the bottom, there are two expandable menu items: "Intro" and "Apps", each with a hamburger menu icon and a downward arrow.

On the right side of the main content area, there is a "Table of contents" link. A grey box is overlaid on the page, containing the following text:

How much can be implemented into <https://annotation.github.io/text-fabric/>?

Or do we need a platform on top of our existing repositories?

5. Knowledge base – but how?



Omeka S Omeka Classic Omeka.net

OMEKA S
For institutions managing a sharable resource pool across multiple sites.
Download v2.1.0 Learn more

OMEKA CLASSIC
For individual projects and educators.
Download v2.7.1 Learn more

How much shall we develop ourselves? And how much can we integrate our resources and knowledge into existing projects?

Title	Creator	Type	Date Added
Jane Austin	Details · Edit · Delete	People	Apr 15, 2017
			Apr 15, 2017
			Apr 13, 2017
		Still Image	Apr 15, 2017
			Apr 15, 2017
The Five Orange Pips	Details · Edit · Delete	Text	Apr 11, 2017

Omeka provides open-source web publishing platforms for sharing digital

6. Workshops





Feature documentation needs update



TF-query examples needed – and need for online storage



Early exposure to the annotated corpora



Tutorials and cookbooks needed – preferably designed by teachers and students



Distilling of knowledge into an easy accessible and dynamic knowledge base



Workshops

Summing up