Pattern for Python 3

Submitted to the Python Software Foundation, September 2014, Tom De Smedt (CLiPS).

Grant proposal

We are asking for funding to make Pattern compatible with Python 3.3.

Pattern (www.clips.uantwerpen.be/pattern and github.com/clips/pattern) is a popular Python 2.7 package for data science [1], with modules whose functions can be intermixed:

**pattern.web** provides an interface to various well-known API’s (Google, Twitter, Facebook, ...) and contains a HTML DOM parser based on BeautifulSoup.

**pattern.db** has a Database wrapper for SQLite and MySQL databases and a Datasheet wrapper that maps a .csv file to a matrix of rows and columns.

**pattern.en** has extensive tools for English natural language processing, such as a text parser that identifies word types (noun, verb, ...), sentiment analysis, modality detection, functions for word inflection, and it comes bundled with WordNet 3. So far, Pattern also has pattern.es, pattern.de, pattern.fr, pattern.nl and pattern.it counterparts for other languages.

**pattern.vector** has tools for machine learning, such as a bag-of-words Model that consists of Document objects, each with a Vector, clustering algorithms, classification algorithms (Naive Bayes, k-NN, SVM, LSA, neural networks) and functions for training and testing (e.g., chi-squared feature selection, gridsearch, confusion matrix, ...).

**pattern.graph** contains functionality comparable to NetworkX for network analysis, and network visualization in the browser using a graph.js helper package.

Other modules include **pattern.server** (a Flask-like interface to CherryPy), **pattern.metrics** (statistical functions) and **canvas.js** (data visualization).

Many of these tools can also be found in separate, more specialized packages such as Scrapy, NLTK, scikit-learn, NetworkX and D3.js. The advantage of Pattern is that everything is uniformly bundled and that it works out-of-the-box. This is interesting for computer science students, computational linguists with little Python experience, and data scientists and web developers that need to combine a lot of different tools under a tight deadline [2].

The package is free for commercial use (BSD-license), well-documented and bundled with dozens of examples and unit tests. It has been featured in the top 50 most popular Python repositories on GitHub since 2012 [3]. Users have called Pattern a "Swiss army knife" and a "treasure trove", and it is in use at universities and companies worldwide.
Grant objective

Data science and technologies such as machine learning and natural language processing are becoming increasingly popular. Today, the World Wide Web is the place of choice to share news, to voice personal opinions or to provide encyclopedic knowledge. Researchers, as well as companies want to harness this vast data set for predictive analytics, for example to uncover cyberbullying on social networks based on language use. The main difficulty is that this data is unstructured: it comes in the form of free text, written in different languages and dialects, with little or no metadata.

Python, alongside R and SAS, is a popular programming language for data science [4] and Pattern is an ideal candidate for teaching purposes or rapid development of data-driven web apps.

Unfortunately, Pattern is also a Python 2.x-only package. It was developed at the Computational Linguistics Research Group (CLiPS) [5] of the University of Antwerp, as part of Tom De Smedt's PhD thesis. Tom is the sole developer of Pattern and nowadays mostly employed on other projects, leaving him with little time to update Pattern to Python 3.

At the same time, the demand for a "Pattern 3" is becoming more urgent. We now receive requests from students and professionals on a weekly basis. We encourage people to help out, but students are usually working on their own topic with little time to spare (or lack experience), and professionals are often not allowed to work on open source projects.

We are hoping to acquire $1,250 funding as an incentive for 1-3 experienced developers to dive in and help us make the source code compatible with Python 3.3. We argue that funding would give developers some peace of mind to write quality source code instead of achieving results as fast as possible. We see the source code as being of equal didactic relevance as the functionality itself. Pattern attempts to follow PEP 8 whenever possible. The source code was also reviewed for quality by the Machine Learning Open Source (MLOSS) community prior to publication in the Journal of Machine Learning Research (JMLR).

CLiPS will contribute an additional $250. Tom De Smedt will contribute an additional $150. The Experimental Media Research Group (EMRG) of the University of Antwerp will contribute an additional $250. We will use available funds to encourage Waylon Flinn to continue his porting work, and attract other experienced Python developers. These can be recently graduated students, developers from the Digital Humanities community or motivated developers recommended through PSF, GitHub or MLOSS [7]. Software developer David Branner has also expressed interest in helping out.
Challenges:

- **Preliminary work**: simpler tasks such as `dict.iteritems()`, `print()`, relative imports and lambda tuple unpacking have already been tackled by Tom.

- **Use six?**: Pattern users have suggested Travis Continuous Integration on GitHub and bundling six as the next two steps. We need to decide if bundling six is the best option. A fork is being worked on by Waylon Flinn [6].

- **Unicode ($800)**: Pattern has tens of thousands of lines of code that intimately deal with text (e.g., web content in various encodings, natural language processing of different languages). Unicode is the biggest challenge, particularly in modules such as `pattern.web` and `pattern.en`.

- **Deprecated stuff ($600)**: `pattern.web` requires special attention because it uses deprecated stuff such as `sgmllib`. It needs to switch between importing BeautifulSoup for Py2 or Py3 etc.

- **Documentation (250$)**: all source code comments, documentation and examples need to be checked and updated. Abundant use is made of `print` which needs to change to `print()`.

- **Refactoring (250$)**: polish up the new code so it is understandable to beginners.

Grant beneficiaries

CLIPS Computational Linguistics Research Group will attract 1-3 good Python developers, selected in mutual agreement with PSF, and closely monitored and tutored by dr. Tom De Smedt, dr. Guy De Pauw and prof. Walter Daelemans of CLIPS.

**Preferred funds delivery**: PayPal account of Tom De Smedt (tomdesmedt@gmail.com).

Grant follow-up

Many tools in Pattern also function as case studies, usually in the form of a blog post detailing the process how they were developed from scratch to a stable package [8] [9]. In the same way, we want to maintain a blog post of the progress in this proposal, i.e., detailing the pitfalls of upgrading a large, Unicode-prone codebase to Python 3.3, which can help and motivate other developers to make the switch.
References


3. GitHub most starred Python repositories (#40 – #50): [https://github.com/search?l=Python&p=5&q=stars%3A%3E1&s=stars&type=Repositories](https://github.com/search?l=Python&p=5&q=stars%3A%3E1&s=stars&type=Repositories)


5. CLiPS Computational Linguistics Research Group: [http://www.clips.ua.ac.be](http://www.clips.ua.ac.be)


