



HAL
open science

OuPoCo, the Combinatorial Poetry Workbench

Thierry Poibeau, Mylène Maignant, Frédérique Mélanie-Becquet, Clément Plancq, Matthieu Raffard, Mathilde Roussel, Mathilde Saurat

► **To cite this version:**

Thierry Poibeau, Mylène Maignant, Frédérique Mélanie-Becquet, Clément Plancq, Matthieu Raffard, et al.. OuPoCo, the Combinatorial Poetry Workbench. Colloque Digital Humanities 2020 (DH2020), Jul 2020, Montréal, Canada. hal-02977154

HAL Id: hal-02977154

<https://hal.science/hal-02977154>

Submitted on 21 Dec 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

OuPoCo, the combinatorial poetry workbench *(L'ouvroir de poésie combinatoire)*

Thierry Poibeau, Mylène Maignant, Frédérique Mélanie-Becquet, Clément Plancq, Matthieu raffard, Mathilde Roussel, Mathilde Saurat

- Laboratoire Lattice

CNRS & École normale supérieure/ PSL & Université Sorbonne nouvelle / USPC

45 rue d'Ulm

75005 Paris

{firstname.lastname@ens.fr}

- Atelier Raffard-Roussel, Paris

raffard.roussel@gmail.com

Oupoco (*L'ouvroir de poésie combinatoire*) is a project taking inspiration from Raymond Queneau's book *Cent mille mille milliards de poèmes*, published in 1961. Queneau's book is a collection of ten sonnets which verses can be freely recombined to form new poems. The book can be seen as composed of ten sheets, each separated into fourteen horizontal bands, each band carrying a verse on its front. The reader can choose, for each verse, one of the ten versions proposed by Queneau. The ten versions of each verse have the same scansion and rhyme, which ensures that each sonnet thus assembled is regular in shape [Queneau, 1961].

It would be tempting to develop a computer-based version of Queneau's work, but Queneau's book is still under copyright, and it is by definition limited to its ten original sonnets. To overcome this problem, we developed the Oupoco project, aiming at proposing a sonnet generator based on the recombination of a large collection of 19th century French sonnets. The challenge is thus more complex than the one proposed originally by Queneau since our sonnets do not have the same scansion and rhyme. From this point of view, even if the project is intended to generate new sonnets, it is largely based on the development of analysis tools able to identify the scansion, the rhyme and the structure of the original sonnets. It is thus very different from the numerous projects dedicated to the pure generation of poetry, being with symbolic [Gervás, P., 2013] or neural methods [Ghazvininejad *et al.*, 2017] [Van De Cruys, 2019] (among many others).

Oupoco is currently based on a collection of 788 sonnets from 16 authors from the 19th century, and this database is regularly expanding. Each sonnet is encoded in a XML format along with related metadata, and a TEI version of the database is available. The project requires to get access to a formal representation of rhymes [Beaudouin, 2002]. In order to do this, the first step is to get a phonetic transcription of the last word of each verse, but this is not enough: for example, *aimé* and *aimée* have the same phonetic transcription, but do not rhyme, according to French rhyming rules (feminine and masculine do not rhyme according to the classical rules of French poetry); there are also cases where the phonetic transcription diverges but words actually rhyme (for example with sounds like [e] and [ɛ]). A series of rules had thus to be defined to get a proper analysis of rhyme from the phonetic transcription of the last word of each verse. The generator uses this analysis to produce random sonnets, with different possible structures, respecting the rules of French

versification (the code and the resources used, especially the sonnet database, are open source and freely available for research, see: <https://github.com/clement-plancq/oupoco-api>).

A series of “side products” have been produced from the project, including:

- a website (<https://oupoco.org/>):
- a bot posting on Twitter (https://twitter.com/oupoco_bot):
- and a “poetry box” (*la boîte à poésie*), a portable version of the original idea that can be demonstrated in public events (based on Raspberry Pi components). Through these devices our goal is to reach a wider audience and engage people to reconnect with poetry.

The main interest of the project is to present French poetry through a new and original setting. With our system, poetry is not any more just a literary genre [Derrida and Ronnel, 1980], but a dynamic object that can be manipulated and experienced. For lots of people, poetry is seen at best as something related to school years, at worse as something boring and uninteresting from the past. Our new setting, in itself, makes it possible to show that playing with poetry can be fun. Our setting puts in perspective the notion of text coherence [Reinhart, 1980], since the result of the generator can be more or less satisfactory from a semantic point of view.

This has two consequences. The first one is related to interpretation: because the machine produces structurally impeccable sonnets, the experienter is unconsciously encouraged to find coherence in them, simply because we are used to coherence in our everyday life and because incoherence is bewildering. The second consequence is a frequent need for the experienter to go back to the original poem, to see where from a given verse originates (tooltips always allows the experienter to go back to the original sonnet where a verse has been extracted). The project is thus not just a sacrilege game over venerated texts, but a way to make people experience and rediscover poetry.

During the presentation, we will detail the technical aspects of the system, and we will examine some examples automatically produced.

References

- Beaudouin, V. (2002). *Mètre et rythme du vers classique. Corneille et Racine*. Champion (coll. Lettres numériques) : Paris.
- Derrida, J. and Ronell, A. (1980). “On Narrative: The Law of Genre”. In *Critical Inquiry*. The University of Chicago Press. 7 (1): 55–81. doi:10.1086/448088.
- Ghazvininejad, M., Shi, X., Priyadarshi, J. and Knight K. (2017). “Hafez: an Interactive Poetry Generation System”. *Proceedings of ACL 2017, System Demonstrations*, Vancouver, Canada. pp. 43–48.
- Gervás, P. (2013). “Computational Modelling of Poetry Generation”, in *Artificial Intelligence and Poetry Symposium, AISB Convention 2013*, University of Exeter, United Kingdom.
- Queneau, R. (1961). *Cent mille milliards de poèmes*. Gallimard : Paris.
- Reinhart T. (1980). “Conditions for Text Coherence”. In *Poetics Today*, Vol. 1, No. 4 (Narratology II: The Fictional Text and the Reader), Duke University Press, pp. 161-180. doi: 10.2307/1771893.
- Van de Cruys, T. (2019). “La génération automatique de poésie en français”. *Proceedings of TALN 2019*, Toulouse, France. pp. 113–126.

Acknowledgement

This work has received support of Translitteræ (Ecole universitaire de recherche, program "Investissements d'avenir" ANR-10-IDEX-0001-02 PSL* and ANR-17-EURE-0025) and the "Investissements d'avenir" program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute).