

On fluid poetry

Eugenio Tisselli

Abstract

Are there potential, non-explicit links between texts on the Web? If we think about the structure of human language as a relational machine, organized according to well-established linguistic relationships between words, then the answer is yes: besides the existence of explicit HTML hyperlinks, it is possible to associate texts through these relationships. However, the actualisation of such immanent links cannot rely on hypertext alone: they need specific processes of transformation in order to become apparent. The apparently loose “text-molecules” need to undergo a process of condensation so that they can come together into a droplet. Text-molecules, which are the main constituent of the Web, can be manipulated and brought together from their remote locations to a client computer by using different operations: a process which can be assimilated to condensation. For instance, text-molecules can automatically be retrieved from distant hypertext documents or databases and translated into other languages using on-line translation services; their words can be substituted by others belonging to the same semantic fields, and so on, in order to join previously separated “molecules” into small “liquid beads” of transformed text: a poetic change of state. In this article, I will use fluid metaphors to describe the inner workings of my piece, “Computer Aided Poetry”: a tool for transforming and manipulating texts which uses remote databases and services, considering the different processes involved in both its English and Spanish versions.

1.

Condensation is the chemical process through which the atoms / molecules of an element or compound in a gaseous state change their aggregation phase into liquid droplets. During condensation, the kinetic energy of the atoms / molecules is reduced by slowing them down. They are brought together as a consequence since the attraction between them, an invisible link, prevails. They also become colder and, because of their aggregation, visible. The droplets formed by the condensation of an unseen gas seem to emerge from out of nowhere. Their sudden mass makes the huddled molecules vulnerable to forces which were latent or too weak in their previous state, such as the pull of gravity, which will make the droplets slide down a smooth surface, leaving behind a liquid trail.

2.

Distillation, a purifying process used since ancient times by winemakers and alchemists, is possible thanks to evaporation and condensation. In distillation, a liquid mixed with solid matter is boiled in order to separate its volatile parts from the solid ones. Then, the vapour is cooled down in special receptacles, a series of serpentine tubes. In these tubes the vapour finally reaches condensation, yielding thus a purified liquid.

3.

Internet is a text-based medium. By this I do not mean that its nature is text-only, but that it is fundamentally made of text. The World Wide Web is, in fact, built on the concept of hypertext: a network of fragmented texts interwoven using explicit links that channel the potentially limitless relationships that can exist among the fragments. Links themselves are made of text, written in HTML language.

When writing on the Web, the sensation of releasing a drop of water into a vast ocean takes physical proportions, transforming the very way in which the writer writes. In fact, most writers nowadays seem to refrain from making the ocean bigger: they don't produce new texts, but point (link) to the writings of others instead. We can see this form of link-writing as the transference of a fluid through a complex,

dynamic network of channels. A network of fluid writing is thus created by using textual, visible pathways between otherwise possibly unconnected ideas. In its “natural” state, this network is not an ocean but a cloud.

4.

“Cloud Computing” is a rather foggy notion, according to which the World Wide Web will be increasingly seen as a platform where not only information, but also different kinds of services and applications will become immediately available, as if coming out of an undifferentiated, nebular space. People will rely less on the software installed on their personal computers, and more on whatever is usable on-line. Despite the possibility that “Cloud Computing” could be no more than a buzzword that can suddenly turn into vapour, it does provide a suggestive image of what the Web is already becoming: a hazy mass of loosely related fragments of information and, importantly, tools for the end-user such as language translators, dictionaries, thesauri or word processors.

5.

Are there potential, non-explicit links between texts on the Web? If we think about the structure of human language as a relational machine, organized according to well-established linguistic relationships between words, then the answer is yes: besides the existence of explicit HTML hyperlinks, it is possible to associate texts through these relationships. However, the actualisation of such immanent links cannot rely on hypertext alone: they need specific processes of transformation in order to become apparent. The apparently loose “text-molecules” need to undergo a process of condensation so that they can come together into a droplet.

Text-molecules, which are the main constituent of the Web, can be manipulated and brought together from their remote locations to a client computer by using different operations: a process which can be assimilated to condensation. For instance, text-molecules can automatically be retrieved from distant hypertext documents or databases and translated into other languages using on-line translation services; their words can be substituted by others belonging to the same semantic fields, and so on, in order to join previously separated “molecules” into small “liquid beads” of transformed text: a poetic change of state.

To illustrate this idea, I will focus on two types of linguistic relationships which are relevant to the work presented here, “Computer Aided Poetry”: synonymy [1] and translation between different languages.

6.

“Computer Aided Poetry” (CAP) is an on-line application [2] built with PHP [3], and defined as “a tool for blocked poets”. CAP allows a writer to enter a phrase and receive a transformed version of it as a result. The words in this new phrase will be different from the original ones, yet they will belong to the same semantic fields. The writer can then modify the resulting phrase and request a new transformation, engaging in a process which is potentially endless. It is precisely this feedback loop, in which the human input is a fundamental element, what differentiates CAP from other algorithmic writing tools, such as phrase generators based on pure randomness or combinatorial techniques. CAP does not have an “intelligent” linguistic algorithm built into it, nor does it rely on local, pre-determined word or phrase banks. What CAP does, much like many of the so-called “Web 2.0” applications, is present the writer with suggestions: “if you entered phrase *A*, you might find phrase *B* interesting”.

CAP works in the following way:

1. The writer enters a phrase in the input field on the application's page. For example, “i love language”. The button labelled “Shake” must be pressed in order to transform the phrase.
2. CAP separates every word in the phrase, and tries to find a synonym for each one of them. In the example above, the word “flame” is obtained as a synonym of “love”, and “tongue” as a synonym of “language”. The subject “i” is left unchanged.
3. The phrase is reconstructed with the proposed synonyms and the unchanged words.
4. After a few seconds, the result is shown on the web page. In this case, the writer sees “i flame tongue”.

At this point, the writer may introduce manual adjustments to the suggested phrase, such as “i flame 'my' tongue”, and request a new transformation. CAP may subsequently respond with something like “i flare my lingo”.

Step number 2 requires a more detailed explanation. As described above, CAP does not have a built-in database of synonyms; instead, it queries a remote server by sending the original word and receiving a list of possible results, if any. More specifically, CAP sends a request to the web server at *thesaurus.reference.com*, which is formed as follows:

<http://thesaurus.reference.com/search?q=???>, where “???” stands for the word for which a synonym must be obtained.

Since the page that the server generates as a result is not available in a machine-readable format, such as XML [4] or JSON [5], a technique called “Screen scraping” must be used to get the synonyms. In short, this technique can be described as a very basic way of parsing a web page, in which the desired data is “manually” subtracted from its surrounding HTML tags [6]. The result of this operation will be a list of synonyms or, more precisely, a set of words which belong to the same semantic field as the original word. CAP will then choose one of them randomly. If no synonyms were found for a particular word, the original one is preserved. Also, words with less than three letters are not considered for the synonym search.

7.

There is a version in Spanish of “Computer Aided Poetry”, called “Poesía Asistida por Computadora” (PAC) [7]. Although the operation of PAC is analogous to that of CAP, there is a very significant difference in its algorithm. The phrase which the writer types in PAC, obviously in Spanish, is translated to English before it is split into words in order to look for synonyms. When the transformed words are reconstructed into a phrase, it is translated back into Spanish and finally shown to the writer. Thus, in the case of PAC, we can expand the steps described above by inserting two new ones:

- 1.5: The phrase is translated from Spanish to English
- 3.5: The phrase is translated from English to Spanish

These translations are obtained by making a request to a remote translation server, *ets.freetranslation.com*.

These requests are done in the following ways:

1.

<http://ets.freetranslation.com/?srctext=???&language=Spanish/English&sequence=core&format=html>, where “???” stands for the phrase that will be translated from Spanish to English (as specified by the

“language” parameter)

2. <http://ets.freetranslation.com/?srctext=???>

&language=English/Spanish&sequence=core&format=html, where “???” stands for the phrase that will be translated from English to Spanish

The results of these requests are retrieved in an HTML format which actually contains no HTML tags but, conveniently, only the translated text.

8.

Since its first implementation in 2006, I have used CAP and PAC as tools for writing poems. I have used PAC more intensively, since my mother language is Spanish. During my writing, I usually follow a simple, recursive method:

1. I enter an initial phrase into PAC. This phrase becomes the title of the poem, and also the original verse for the first transformation.
2. When a transformed phrase appears on the screen, I limit my intervention to correcting its grammar. Sometimes I will add or discard one or two words. The new, adjusted phrase becomes then a new line of the poem.
3. I request a subsequent transformation of this verse, and return to step 2 until I consider that the poem is complete.

The decisions I make during the process are purely grammatical and aesthetic; they never have to do with the semantic content of the verses. I also decide when to stop the process and complete the poem. This method reveals a personal pursuit: to write without the weight of intentions, goals or meanings, and yet without excluding the writer from the process (as is often the case in automatic text generators)

I consider the poems I have written using PAC to be distilled texts, obtained from a transitional, looping process of evaporation (phrases are decomposed into molecules and dispersed in the network cloud) and condensation (new phrases that seem to emerge “from out of nowhere”, formed by multiple text-molecules which are gathered from the cloud and brought together)

Fluidity lies at the core of my way of using CAP and PAC. I consider intentions, goals and meanings as heavy, solid matter from which I wish to liberate my writing. These tools, which treat words and phrases like fluids that can go through state changes, allow me to separate solid from liquid, to control the path that this process will follow, and to choose when to stop it. As a writer, I become a proofreader of the computer's output, and a designer of the final form of the poem. By making myself *available*, I become a channel through which language flows.

Most of the poems I write seem absurd on a first reading, and yet that is precisely the point. They are like a porous surface with plenty of holes, which the reader must fill with another fluid: her own interpretation. This sense-making task assigned to the willing reader can be supported by the fact that the verses maintain a certain semantic coherence throughout a poem. Below the poems' nonsensical appearance, there lies an implicit link that bonds the verses together, thanks to the relations of synonymy which extend from the words of one verse to the next.

9.

The heart of the question is this: for us, the world is reality; for the machines, the only thing that is truly real is language - Stanislaw Lem [8]

10.

In May 2010, the book “El drama del lavaplatos”, which contains poems I wrote using PAC, was published [9]. As described above, my role as a writer is very different from the traditional poet: instead of searching for the precise words for expressing my ideas, thoughts or feelings, I simply accepted and corrected the computer's recursive distillations of the text.

Interestingly, CAP and PAC can be used in alternative ways, such as the approach taken by the Spanish writer Vicente Luis Mora. In his experience with PAC, he tried to coax the tool into generating a verse which could be considered as being similar in style to the verses written by Luis de Góngora, the Spanish Baroque poet. Mora describes his search in an article he wrote for the on-line magazine “Frontera D” [10]. In his article he claims that, after a certain number of transformations, he obtained a couple of verses which could have been penned by Góngora himself.

The algorithms in CAP and PAC have also been used in other works and applications. In some of my works, such as “Synonymovie” (2004), “La mareadora” (2005), or “JB Wock” (2006), variations of the algorithm are used [11]. In some cases, the algorithm is also complemented by image searches. More recently, I have applied this potential for transforming and channelling texts to my work “Facebots” (2009-2010), in which three artificial agents automatically and periodically publish posts on their Facebook profiles [12].

11.

The Uruguayan writer Felisberto Hernández said that he found joy in being friends with words, and that he was truly delighted when two of them, “which became attracted in an unwatched place of my soul”, came together for the first time [13]. Substitute “my soul” for “the Internet” in Felisberto's phrase. CAP intends to be a facilitator of joyous discoveries; a tool for revealing fresh and unsuspected liaisons among words. It was built to be a mechanism for surprise in the age of fluid text.

Joy, even laughter. In a fragment of David Clark's digital work “88 constellations for Ludwig Wittgenstein” [14], two friends talk about philosophy over coffee:

He said: “the limits of language are the limits of my world. And by speaking, I limit the world. That's Wittgenstein. Wittgenstein also said: our words will only express facts, as a teacup will hold a teacup full of water, even if I were to pour out a gallon over it”

And then she laughed out loud, and she said: “Laughter is the limits of language! We laugh when the absurdity of language becomes apparent! When it tricks us into believing in a thing called meaning”

12.

... to pour endless drops of text into our teacup. It will overflow, unless the cup has a hole in its bottom.

NOTES:

1.

A synonym is a word having the same or nearly the same meaning as another word or other words in a language. While the more accurate and refined term “semantic field” should be used to describe words which belong into the same group because of their meaning, “synonymy” is preferred here simply because of its usability.

2.

<http://motorhueso.net/cap>

3.

PHP (Hypertext Preprocessor) is a server-side scripting language, used mostly to create dynamic web pages. <http://www.php.net>

4.

XML (Extensible Markup Language) is a user-defined set of tags for encoding electronic documents. It provides a way to make information machine-readable, and thus facilitates the exchange of content between web sites.

5.

JSON (JavaScript Object Notation) is a format for data exchange oriented towards machine-readability.

6.

The process of Screen-scraping involves a number of different steps:

1. Identify the exact area inside the HTML code in which the data to be extracted is found.
2. Isolate the identified area from the rest of the web page's code.
3. Strip this fragment of its HTML tags.

For example, the result page of the synonym server will contain the synonyms encoded as follows:

```
<td valign="top">Synonyms:</td>
<td><span>
<a class="theColor" rel="nofollow" href="http://thesaurus.com/browse/adulation">adulation</a>,
<a class="theColor" rel="nofollow" href="http://thesaurus.com/browse/affection">affection</a>,
<a class="theColor" rel="nofollow" href="http://thesaurus.com/browse/allegiance">allegiance</a>,
<a class="theColor" rel="nofollow" href="http://thesaurus.com/browse/amity">amity</a>,
<a class="theColor" rel="nofollow" href="http://thesaurus.com/browse/amour">amour</a>,
...
</span></td>
```

The synonyms are found between the “Synonyms:” and “” delimiters. Stripping this fragment of code from its HTML tags will result in a list of synonyms for the word “love”: “adulation affection allegiance amity amour etc.”

Needless to say, Screen scraping can be considered as a very raw and unreliable technique for data extraction. Its main drawback is that it is very sensitive to changes in the HTML encoding of a web page: if the encoding is modified in the remote server, the scraping algorithm must also be modified accordingly. Fortunately, the encoding of the synonym server has remained unchanged since the first implementation of CAP in 2006. However, periodical checks are necessary to make sure that the algorithm is still functional.

7.

<http://motorhueso.net/pac>

8.

This phrase is a translation from the Spanish version of Stanislaw Lem's short story, “Historia de la Literatura Bítica en Cinco Volúmenes”, included in the book “Magnitud Imaginaria”. Editorial

Impedimenta, Salamanca, Spain, 2010.

9.

Tisselli, Eugenio “El drama del lavaplatos”. Salamanca, Spain. 2010.

10.

Mora, Vicente Luis; Chiappe, Domenico; Tisselli, Eugenio “Góngora asistido”.
<http://www.fronterad.com/?q=node/789>

11.

All of these works are available at <http://motorhueso.net>

12.

<http://motorhueso.net/facebook>

13.

The words of Felisberto Hernández were taken from his web site:
<http://www.felisberto.org.uy/literatura.html>

14.

Clark, David. “88 constellations for Ludwig Wittgenstein”, <http://www.88constellations.net/>