O Poeta Artificial **2.0:**
Increasing Meaningfulness in a Poetry Generation Twitter bot

Hugo Gonçalo Oliveira
CISUC, Department of Informatics Engineering
University of Coimbra, Portugal
`hroliv@dei.uc.pt`

Abstract

O Poeta Artificial is a bot that tweets poems, in Portuguese, inspired by the latest Twitter trends. Built on top of PoeTryMe, a poetry generation platform, so far it had only produced poems based on a set of keywords in tweets about a trend. This paper presents recently implemented features for increasing the connection between the produced text and the target trend through the reutilisation and production of new text fragments, for highlighting the trend, paraphrasing text by Twitter users, or based on extracted or inferred semantic relations.

1 Introduction

Poetry generation is a popular task at the intersection of Computational Creativity and Natural Language Generation. It aims at producing text that exhibits poetic features at formal and content level, while, to some extent, syntactic rules should still be followed and a meaningful message should be transmitted, often through figurative language. Instead of generating a poem that uses a set of user-given keywords or around an abstract concept, several poetry generators produce poetry inspired by a given prose document. Besides the potential application to entertainment, this provides a specific and tighter context for assessing the poem's interpretability.

This paper presents new features of *O Poeta Artificial* (Portuguese for "The Artificial Poet"), a computational system that produces poems written in Portuguese, inspired by the latest trends on the social network Twitter and, similarly to other creative systems, posts them in the same network, through the *@poetartificial* account. *O Poeta Artificial* is built on top of PoeTryMe (Gonçalo Oliveira, 2012),

a poetry generation platform, and originally used the latter for producing poetry from a set of frequent keywords in tweets that mentioned the target trend. *O Poeta Artificial* 2.0, hereafter shortened to *Poeta* 2.0, resulted from recent developments on the original version, aimed at increasing the interpretability of its results through a higher connection with the trend. The new features enable the reutilisation of fragments of human-produced tweets, possibly with a word replaced by its synonym, as well as the inclusion of fragments that highlight the trend, or fragments obtained from relations extracted from tweets about the trend, or even inferred from the latter. Produced poems may include some new fragments and others produced by the original procedure (hereafter, the classic way), based on the extracted keywords, while keeping a regular metre and favouring the presence of rhymes.

The remainder of this paper starts with a brief overview on poetry generation systems and creative Twitter bots, followed by a short introduction to PoeTryMe and how it is used by the Twitter bot. After this, the new features of *Poeta* 2.0 are described, with a strong focus on new kinds of fragments produced by this system. Before concluding, the results of *Poeta* 2.0 are illustrated with some poems produced, using different kinds of fragments.

2 Related Work

Automatic poetry generators are a specific kind of Natural Language Generation (NLG) systems where the produced text exhibits poetic features, such as a pre-defined metre and rhymes, together with some level of abstraction and figurative language.

Many poetry generators have been developed and described in the literature (see Gonçalo Oliveira (2017b)), especially in the domain of Computational Creativity. They are typically knowledge-intensive intelligent systems that deal with several levels of language, from lexical choice to semantics.

Several systems produce poetry based on given stimuli, which can be a set of semantic predicates (Manurung, 2003), one (Gonçalo Oliveira, 2012) or more seed words (Gonçalo Oliveira, 2012), a prose description of a message (Gervás, 2001), or a longer piece of text, such as blog posts (Misztal and Indurkhya, 2014) or newspaper articles (Colton et al., 2012; Rashel and Manurung, 2014; Toivanen et al., 2014; Tobing and Manurung, 2015; Gonçalo Oliveira and Alves, 2016). Longer documents used as inspiration can be reflected in the poems through the use of keywords (Rashel and Manurung, 2014), associations (Toivanen et al., 2014), phrases (Charnley et al., 2014), similes (Colton et al., 2012), dependency (Tobing and Manurung, 2015) or semantic relations (Gonçalo Oliveira and Alves, 2016) extracted from them, and may also transmit the same sentiment (Colton et al., 2012) or emotions (Misztal and Indurkhya, 2014). Poems are typically built from templates, either handcrafted or extracted from human-produced poems, then filled with information from the inspiration document.

Twitter has become a popular platform for bots, mostly because of its nature – many users posting short messages (tweets), available on real time – and its friendly API, which exposes much information, easily used by computational systems. This is also true for creative bots. Some use Twitter merely as a showcase for exhibiting their results, possibly enabling some kind of user interaction, liking or retweeting. Those include, for instance, bots for producing riddles (Guerrero et al., 2015) or Internet Memes (Gonçalo Oliveira et al., 2016). Other bots also exploit information on Twitter for producing their contents. This happens, for instance, for @*poetartificial* (Gonçalo Oliveira, 2016), which produces poetry, in Portuguese, roughly inspired by current trends, and is the focus of the following sections. It is also the case of @*Metaphor-Magnet* (Veale et al., 2015) and its "brother" bots, who produce novel metaphors, through the same generating mechanisms as a poetry generation sys-

tem (Veale, 2013), more focused on content and not so much on form.

Despite the growing number of intelligent bots, Twitter has many other bots, some of which performing tasks that are typically in the domain of creativity, but through not so intelligent and knowledge-poor processes. Those include @*MetaphorMinute*, which generates random metaphors, or @*pentametron*, which retweets pairings of random rhyming tweets, both with ten metrical syllables.

Besides bots, other creative systems produce content inspired by information circulating on Twitter, including poetry. FloWr (Charnley et al., 2014) is a platform for implementing creative systems, which has been used for producing poetry by selecting phrases from human-produced tweets, based on sentiment and theme, and organising them according to a target metre and rhyme. TwitSonnet (Lamb et al., 2017) builds poems with tweets retrieved with a given keyword in a time interval, scored according to the poetic criteria of: reaction (presence of words that transmit a desired emotion), meaning (presence of given keywords and frequent tri-grams), and craft (metre and rhyme, plus words with strong imagery). Several poems by TwitSonnet were posted on Tumblr, another micro-blogging social network. Instead of templates, the previous systems reuse complete text fragments extracted from Twitter.

3   PoeTryMe and *O Poeta Artificial*

O Poeta Artificial (Gonçalo Oliveira, 2016) is a Twitter bot that tweets poems written in Portuguese and inspired by recent trends in the Portuguese Twitter community. It is built on top of PoeTryMe (Gonçalo Oliveira, 2012), a poetry generation platform with a modular architecture, so far adapted to produce poetry in different languages and from different stimuli.

PoeTryMe's architecture, explained in detail elsewhere (Gonçalo Oliveira et al., 2017), is based on two core modules – a Generation Strategy and the Lines Generator – and some complementary ones. To some extent, a parallelism can be made between this architecture and the traditional 'strategy' and 'tactical' components of a NLG system (Thompson, 1977). The Generation Strategy implements a plan

for producing poems according to user-given parameters. It may have different implementations and interact with the Syllable Utils for metre scansion and rhyme identification. The Lines Generator interacts with a semantic network and a context-free grammar for producing semantically-coherent fragments of text, to be used as lines of a poem. Each of those lines will generally use two words that, in the semantic network, are connected by some relation R. Those words fill a line template, provided by the grammar, which is generalised to suit all pairs of words related by R. For instance, the line template `you're the X of my Y` can be used for rendering partOf relations, such as:

- estuary partOf river
  → *you're the estuary of my river*
- periscope partOf submarine
  → *you're the periscope of my submarine*
- fiber partOf personality
  → *you're the fiber of my personality*

In most instantiations of PoeTryMe, a set of seed words is provided as input for setting the poem domain. This constrains the semantic network to relations that involve one of the seeds, with a probability of selecting also relations with indirectly related words (known as the 'surprise factor'). There is also a module for expanding the set of seeds with structurally-relevant words, possibly constrained by a target polarity (positive or negative). Though originally developed for Portuguese, poetry may also be generated in Spanish or English, depending on the underlying linguistic resources, namely the semantic network, the lexicons and the grammars (Gonçalo Oliveira et al., 2017).

O Poeta Artificial adds an initial layer for selecting the seed words to use. Before generation, it: (i) Selects one of the top trends in the Portuguese Twitter (the highest not used in the last three poems); (ii) Retrieves recent tweets (currently, up to 200), written in Portuguese and mentioning the target trend; (iii) Processes each tweet and extracts every content word used; (iv) Selects top frequent content words (currently, 4) to be used as seeds; (v) May expand the seeds, either according to the main sentiment of the tweets (based on the presence of emoticons) or, if there is a Wikipedia article about the trend, with content words from its abstract. PoeTryMe is then used for producing 25 poems from

the seeds, following a generate-and-test strategy at the line level. The poem with the highest score for metre and presence of rhymes is tweeted.

In the original version of *O Poeta Artificial*, the result was always a block of four lines, generally with 10 syllables each, and with occasional rhymes. Due to their generation process, lines were syntactically-correct and semantically coherent, but the connection with the trend was often too shallow. For instance, as the trend is typically a hashtag or a named entity, it is not in the semantic network and thus never used in the poem. The following section describes recent developments towards a higher connection with the trend, thus contributing to an improved meaningfulness.

4 *Poeta Artificial* 2.0: beyond seed words

Poeta 2.0 aims at improving the meaningfulness of the original bot by increasing the connection of the produced poems with the target trend and with what people are saying about it.

A minor improvement occurs in the seed selection process. Instead of relying exclusively on the frequency of each content word in the tweets, *Poeta* 2.0 divides it by its frequency in a large Portuguese corpus (Santos and Bick, 2000). This aims to use more relevant words, and can be seen as an application of the *tf.idf* weighting scheme.

Yet, the main feature of *Poeta* 2.0 is that, besides seed words, it also provides a set of pre-generated text fragments to PoeTryMe, somehow connected to the target trend and that may be used as poem lines. For every line of the poem to fill, there is a probability of using one of the generated fragments instead of a line produced in the classic way, based on the semantic network and generation grammar. This probability is proportional to the number of fragments of this kind available for the target number of syllables. One of the previous fragments is also used if it has exactly the target number of syllables and rhymes with one of the previously used lines.

Another new feature is that, based on the produced fragments, *Poeta* 2.0 sets the target length of the poem lines, though having in mind the maximum of 140 characters a tweet can contain. More precisely, it counts the number of syllables of each text fragment produced and selects a number, between 5

and 10, for which there are more fragments available. Poems by *Poeta* 2.0 are still blocks of four lines, but each line will have the selected number of syllables or close.

The remainder of this section describes the different types of text fragments that *Poeta* 2.0 produces, namely fragments that highlight the trend, fragments of the processed tweets, paraphrases of the former, and fragments based on semantic relations. All are put together in a set of usable fragments. PoeTryMe will have no idea of how they were produced.

4.1 Fragments Highlighting the Trend

The first kind of fragments is based on a small set of templates with a placeholder for the target trend, each highlighting the latter by referring to it as a recent topic that many people are talking about. Some of those templates are shown in table 1, where T is the trend placeholder.

andam a escrever/falar sobre T
(they are writing/talking about T)
hoje fala-se de T
(today, people are talking about T)
sobre T vim escrever
(about T I came to write)
interesse por T é global?
(interest for T is global)
T é tendência social
(T is a social trend)
T é um assunto recente
(T is a recent topic)
fala de T muita gente!
(many people chatting about T)
T, porque falam de ti?
(T, why do they chat about you?)
T, T, e T
(T, T, and T)

Table 1: Examples of trend-highlighting templates.

4.2 Fragments of Tweets

Similarly to other systems (Charnley et al., 2014; Lamb et al., 2017), *Poeta* 2.0 may reuse text from human-produced tweets. Recall that, in order to select the most relevant words for the target trend, 200 tweets written in Portuguese and mentioning this trend are used as an inspiration set. Among the processing steps, those tweets are broken into smaller units, when possible, following simple rules, such as line breaks or punctuation signs. Each of the obtained units is added to the set of fragments

Original fragment:
Salvador com dúvidas em aceitar
(Salvador with doubts whether to accept)
Synonyms:
dúvidas = {indecisões, hesitações, incertezas}
(doubts = {indecisions, hesitations, uncertainties}))
aceitar = {aprovar, acatar, adoptar}
(accept = {aprove, obey, adopt})
Paraphrases:
Salvador com indecisões em aceitar
Salvador com hesitações em aceitar
Salvador com incertezas em aceitar
Salvador com dúvidas em aprovar
Salvador com dúvidas em acatar
Salvador com dúvidas em adoptar

Table 2: Tweet and some generated paraphrases.

provided to PoeTryMe. The main difference between *Poeta* 2.0 and other poetry generators that use human-written tweets is that *Poeta* 2.0 mixes them with the other kinds of fragments it produces.

4.3 Paraphrases of Tweets

Besides human-written tweets, *Poeta* 2.0 produces variations of them. More precisely, it retrieves synonyms of the content words in the previous fragments from PoeTryMe's semantic network, and produces new fragments by replacing each content word with one of its synonyms. *Poeta* 2.0 may thus find alternative ways of expressing the same messages humans did, possibly also covering a wider range of metres. This has similarities with Tobing and Manurung (2015), though *Poeta* 2.0 does not perform word sense disambiguation because PoeTryMe's semantic network is not organised in word senses. Although some issues may result from ambiguity, we prefer to think that, though not completely intentional, using synonyms that only apply for other senses may create interesting domain shifts. Table 2 illustrates this procedure for a specific fragment.

In order to avoid poems where all lines paraphrase each other, a maximum of 5 paraphrases are generated for each content word in a fragment. If a word has more than 5 synonyms, 5 are randomly selected.

4.4 Semantic Relation-based Fragments

In order to keep the philosophy behind PoeTryMe, the natural way of increasing interpretability would be to extract semantic relations from the tweets mentioning the trend and adding them to the set of relations to use. To some extent, we kept this philoso-

phy, but we also wanted *Poeta* 2.0 to be independent from the core of PoeTryMe. This enables the extraction of relations of different types, more focused on Twitter text, on the trends, and possibly not so well-defined, which can be managed without changing PoeTryMe. The same happens for a new set of line templates based on the extracted relations, smaller but more controlled than the line templates covered by PoeTryMe's grammars, most of which acquired automatically from collections of poetry.

Another important reason for this decision is that, in Portuguese, determiners, adjectives and other words are declined according to gender and number. In PoeTryMe, this is handled by a morphology lexicon and different grammar productions are still required, depending on the gender and number of the related words. Yet, while the same lexicon could be used for acquiring the gender and number of nouns or adjectives extracted from Twitter, it would not cover all the trends, which are typically named entities, or hashtags. Therefore, the templates for the Twitter relations are, as much as possible, gender and number independent, and only consider these attributes when they can be obtained from PoeTryMe.

In order to produce text fragments based on semantic relations involving the trend, *Poeta* 2.0 relies primarily on a small set of line templates compatible with each of the covered semantic relations. Yet, it goes further and combines the extracted relations with the relations in PoeTryMe's semantic network for inferring new relations and increasing, once again, the set of available fragments. The following sections describe the three steps involved in the production of relation-based fragments: extraction, inference, and text generation.

4.4.1 Relation Extraction

Since Hearst (1992) proposed a set of lexical-syntactic patterns for the automatic acquisition of hyponym-hypernym pairs from text, much work has targeted the automatic extraction of semantic relations from text, sometimes with much more sophisticated approaches. Yet, when recall is not critical, one of the arguments is fixed (the trend), and we are focused on a closed set of relation types, relying on a small set of lexical-syntactical patterns is probably the fastest way for achieving this goal. Moreover, it avoids the need for large quantities of en-

coded knowledge and provides higher control on the results than for machine learning approaches.

Currently, four different relation types are extracted from the inspiration tweets. This is performed with the help of a small set of patterns, revealed in table 3 and with possible results illustrated in table 4. In both tables, T stands for the trend, and a rough translation of the patterns, from Portuguese to English, is provided.

The extracted relations – isA, hasProperty, has, can – are tied to the extraction patterns but are not as semantically well-defined as relations in a wordnet or ontology. Yet, as long as we are aware of this in the following steps, it is not a critical issue.

4.4.2 Relation Inference

Based on the extracted relations, implicit in the text, other relations are inferred, when combined with relations in PoeTryMe's semantic network. For Portuguese, the network currently used includes all the relations in at least two out of nine Portuguese lexical-semantic knowledge bases, including wordnets and dictionaries (Gonçalo Oliveira, 2017a). Therefore, it covers a rich set of relation types including not only synonymy, hypernymy and partOf, but also others, such as isSaidOfWhatDoes (in Portuguese, *dizSeDoQue*), isSaidAbout (*dizSeSobre*), hasQuality (*temQualidade*), hasState (*temEstado*), antonymyOf (*antonimoDe*), isPart/Member/MaterialOf (*parte/membro/materialDe*), and isPartOfWhatIs (*parteDeAlgoComPropriedade*), which are exploited by *Poeta* 2.0

A set of rules was handcrafted for inferring new relations from a combination of one relation extracted from the tweets and another in PoeTryMe's semantic network. Although more inference rules may be defined in the future, possibly exploiting additional relations, the current rules are in figure 1. Again, the inferred relations are not as well-defined as those in a wordnet. Some are of the same types as the relations originally extracted, but new types are introduced (e.g. isLike, isNot, withQuality, withState, mayCause), some of which may result in metaphors or less obvious connections, and are thus useful for poetry generation. Table 5 illustrates some of the previous rules with examples of relations extracted, known (i.e. in PoeTryMe's semantic network), and inferred.

Pattern	Relation
`... <T> (é|parece) (o|a|um|uma) <N> ...` *(T is a/the N)*	T isA N
`... <T> (é|parece) <ADV> <ADJ> ...` *(T is/seems ADJ)*	T hasProperty ADJ
`... tão <ADJ> (como|quanto) (o|a|um|uma)? <T> ...` *(as ADJ as a/the? T)*	T hasProperty ADJ
`... <T> está <ADJ> ...` *(T is ADJ)*	T hasProperty ADJ
`... <T> tem <N> ...` *(T has N)*	T has N
`... <T> (está)? a <V> ...` *(T is V-ing)*	T can V
`... <V> como (o|a|um|uma)? <T> ...` *(V like/as a/the? T)*	T can V

Table 3: Patterns considered and extracted relations.

Text	Relation
Bruno Mars é o rei do pop. *(Bruno Mars is the king of pop.)*	*Bruno Mars* isA *rei* *(Bruno Mars* isA *king)*
O Centeno é mesmo brilhante... *(Centeno is really brilliant...)*	*Centeno* hasProperty *brilhante* *(Centeno* hasProperty *brilliant)*
Wagner Moura foi tão sincero quanto Lula. *(Wagner Moura was as sincere as Lula.)*	*Lula* hasProperty *sincero* *(Lula* hasProperty *sincere)*
O António Costa está feliz da vida! *(António Costa is happy of his life!)*	*António Costa* hasProperty *feliz* *(António Costa* hasProperty *happy)*
Lorde tem talento demais. *(Lorde has too much talent.)*	*Lorde* has *talento* *(Lorde* has *talent)*
Manuel Serrão a pensar exatamente o mesmo que eu. *(Manuel Serrão is thinking exactly the same as I.)*	*Manuel Serrão* can *pensar* *(Manuel Serrão* can *think)*
Cantar como a Adele é dificílimo! *(To sing like Adele is so hard!)*	*Adele* can *cantar* *(Adele* can *sing)*

Table 4: Examples of extracted relations.

Extracted	Known	Inferred
T isA *rei* (king)	*real* (royal) isSaidAbout *rei*	T hasProperty *real*
T hasProperty *brilhante* (brilliant)	*brilhante* isSaidAbout *luminosidade* (light) *brilhante* hasQuality *brilhantismo* (brilliance)	T isLike *luminosidade* T isLike *brilhantismo*
T hasProperty *sincero* (sincere)	*sincero* hasQuality *sinceridade* (sincerity) *sincero* antonymOf *hipócrita* (hipocrit)	T withQuality *sinceridade* T isNot *hipócrita*
T has *talento* (talent)	*capaz* (capable) saidAbout *talento* *talento* isPartOfWhatIs *talentoso* (talented)	T is *capaz* T is *talentoso*
T can *pensar* (think)	*pensante* (thinker) saidOfWhatDoes *pensar* *pensar* causes *pensamento* (thought)	T is *pensante* T mayCause *pensamento*

Table 5: Examples of inferred relations.

4.4.3 Semantic Relations as Text

Both extracted and inferred relations are used for producing text fragments by filling, with the relation arguments, a small set of handcrafted templates, compatible with each relation type. Table 6 illustrates this with examples of fragments produced for a set of relations. Some fragments use both relation arguments, while others only use the second argument, and not the trend, to avoid much repetition.

5 Examples

This section presents some poems produced by *Poeta* 2.0, their rough English translations, and a short discussion on the fragments used. Despite the new features introduced, sometimes, poems still have all of their lines generated in the classic way. This happens especially when no tweets are reused, possibly due to their long size, and when no relations are extracted. The following poem is of this kind:

Relation	Example fragments	
Bruno Mars isA *rei*	*ser <u>rei</u> como <u>Bruno Mars</u>*	(being a king like Bruno Mars)
	por ser <u>rei</u>	(for being a king)
Centeno hasProperty *brilhante*	*quero ser <u>brilhante</u> como <u>Centeno</u>*	(I want to be brilliant like Centeno)
	dizem que é <u>brilhante</u>	(people say he's brilliant)
Lorde has *talento*	*<u>Lorde</u> tem <u>talento</u>*	(Lorde has talent)
	tem mesmo <u>talento</u>!	(she really has talent!)
Adele can *cantar*	*a <u>cantar</u> como <u>Adele</u>*	(singing like Adele)
	dizem que sabe <u>cantar</u>!	(people say she knows how to sing!)
Centeno isLike *luminosidade*	*<u>Centeno</u> lembra a <u>luminosidade</u>*	(Centeno resembles brightness)
	com uma <u>luminosidade</u> tal	(with such a brightness)
Lula withQuality *sinceridade*	*a <u>sinceridade</u> do <u>Lula</u>*	(Lula's sincerity)
	a demonstrar <u>sinceridade</u>	(showing sincerity)
Lula isNot *hipócrita*	*<u>Lula</u> não será <u>hipócrita</u>*	(Lula is probably not a hipocrit)
	nada <u>hipócrita</u>	(not a hipocrit)
Manuel Serrão is *pensante*	*<u>Manuel Serrão</u> parece <u>pensante</u>*	(Manuel Serrão seems to be a thinker)
	também quero ser <u>pensante</u>	(I also want to be a thinker)
Manuel Serrão mayCause *pensamento*	*como o <u>pensamento</u> de <u>Manuel Serrão</u>?*	(like Manuel Serrão's thought?)
	a gerar <u>pensamento</u>	(generating a thought)

Table 6: Relations and examples of produced fragments.

- If (T isA X) $\wedge$

    X isSaidOfWhatDoes Y $\rightarrow$ T can Y

    Y saidAbout X $\rightarrow$ T is Y

    X hasQuality Y $\rightarrow$ T withQuality Y

    X hasState Y $\rightarrow$ T withState Y

    X antonymOf Y $\rightarrow$ T isNot Y

    Y isPart/Member/MaterialOf X $\rightarrow$ T has Y

- If (T hasProperty X) $\wedge$

    X isSaidOfWhatDoes Y $\rightarrow$ T can Y

    X isSaidAbout Y $\rightarrow$ T isLike Y

    X hasQuality Y $\rightarrow$ T withQuality Y

    X hasState Y $\rightarrow$ T withState Y

    X antonymOf Y $\rightarrow$ T isNot Y

    Y hasQuality X $\rightarrow$ T isLike Y

    Y hasState X $\rightarrow$ T isLike Y

- If (T has X) $\wedge$

    Y isSaidAbout X $\rightarrow$ T is Y

    Y hasQuality X $\rightarrow$ T is Y

    Y hasState X $\rightarrow$ T is Y

    X isPartOfWhatIs Y $\rightarrow$ T is Y

- If (T can X) $\wedge$

    Y isSaidOfWhatDoes X $\rightarrow$ T is Y

    X causes Y $\rightarrow$ T mayCause Y

Figure 1: Rules for relation inference

delatar sempre causa delação — To denounce always causes denunciation
delação negra sem acusação — Black denunciation without accusation
acusação em meia delação — Accusation in half denunciation
sem achar cita, nem citação — Without quotation or citation

It was generated for the trend *Carlos Alexandre*, the name of a Portuguese judge in charge of several cases with great public impact. All lines rhyme and all have 10 syllables, except the last, which has only 9. The seeds collected from the tweets were *delação* (denunciation), *advogada* (lawyer), *telefónica* (of telephone), *cita* (citation). The first line was produced from the semantic relation '*delatar* causes *delação*', the second and third from '*acusação* synonymOf *delação*', and the fourth from '*citação* synonymOf *cita*'.

The following example was produced on the morning of 4th of June 2017, after the attacks at London Bridge, when there was a trending hashtag *#LondonBridge*:

fala de #LondonBridge muita gente! — Many people talking about #LondonBridge!
O universo é mesmo doente — The universe is really sick
Polícia procura suspeitosos — Police searching for suspects
Polícia procura duvidosos — Police searching for dubious

All the lines have 10 syllables, except the first, because the syllable division tool considered the # as a syllable. Every line ends in rhyme: the first pair of lines ends in *-ente* and the second in *-osos*. The first line highlights the trend and the remaining are paraphrases of the following fragments from human-written tweets:

O mundo é mesmo doente (The world is really sick)

Polícia procura suspeitos (Police looking for suspects)

The next example was produced for the trend *Rui Santos*, a Portuguese football commentator, two days after Benfica won the Portuguese Football

Cup (30th May 2017). It uses three lines based on relations extracted from the processed tweets:

> Rui Santos consegue falar também posso ser miliar? também quero ser miliar seboso a par, par a par

Rui Santos can speak can I be very small as well? I also want to be very small greasy at hand, parwise

All of its lines have 8 syllables and all have the same termination. The first line was produced from the relation '*Rui Santos* can *falar*' (talk), extracted from more than one tweet, including the following:

No lugar do Rui Vitória deixava o seboso do Rui Santos a falar sozinho com o seu troféu.
(If I were Rui Vitória, I would leave greasy Rui Santos talking alone with his trophy)

Another relation is '*Rui Santos* has *dimensão*' (dimension), extracted from:

Rui Santos tem dimensão para o Sporting. É pequenino.
(Rui Santos has dimension for Sporting. He is little.)

The second and third lines of the poem were produced from the relation '*Rui Santos* is *miliar*' (very small), inferred from the previous, due to the relation '*dimensão* isPartOfWhatIs *miliar*'.

The final example was produced for the trend *Ronaldo*, one day after the football player Cristiano Ronaldo won the fourth European Champions League of his career (5th June 2017). It mixes different kinds of fragments:

> Ronaldo é muito falado arte e dança amor calado num estado de felicidade Ronaldo mostra simplicidade

Ronaldo is widely spoken art and dance silent love in a state of happiness Ronaldo shows simplicity

All lines have 9 syllables, with two rhymes: the first pair ends in *-ado* and the second in *-ade*. The first line highlights the trend. Due to a video of Ronaldo dancing, one of the seeds extracted was *dança* (dance), which originated the second line, based on the relation '*arte* hypernymOf *dança*'. The remaining lines result from two relations: '*Ronaldo* withQuality *simplicidade*' (inferred from '*Ronaldo* hasProperty *feliz*' and '*feliz* hasState *felicidade*'), and '*Ronaldo* withQuality *simples*' (inferred from '*Ronaldo* hasProperty *simples*' and '*simples* hasQuality *simplicidade*').

6 Concluding Remarks

In order to increase the connection between poems by a Twitter bot and a recent trend, more meaningful text fragments are now produced and, when pos-

sible, used in the poems. This paper described the production of those fragments.

The first impression of the poems now generated is positive, which is also shown by the examples included in this paper. Some poems are still produced in the classic way, where the only connection between lines and trend is the presence of associated words in semantically-coherent sentences. Yet, several have now lines that highlight the trend, lines that are built from relations involving the trend, or lines that reuse text by other users about the trend, thus making them more meaningful. Each kind of fragments may be further augmented, for instance, by exploiting additional patterns and semantic relations in the tweets, but the manual labour involved is a practical issue, as it may become quite complex to manage all the patterns and inference rules.

Another limitation is that the semantic relation-based fragments have to be gender and number independent. This may be minimised in the future, if the determiners frequently used before the trends are considered for identifying the previous properties. Yet, as there are other kinds of fragments, other relations, and poems only have four lines, this is currently not critical.

Most limitations of PoeTryMe (Gonçalo Oliveira et al., 2017) are also present. For instance, despite targeting the same semantic domain, lines are generated independently of each other, not always resulting in the most logical sequence. This could be minimised if a reordering procedure was applied, similar to the one by Lamb et al. (2017), where abstraction and imagery are considered.

The extraction of long-term information on the trend may also be improved. Currently, if the trend has a Wikipedia article, associations are extracted from its abstract. In the future, relations may be extracted directly from DBPedia.

A final issue, not yet discussed, is that the system may reuse fragments that contain typos, thus decreasing the quality of the poems. Of course, every word could be spellchecked and words with typos could be corrected, possibly to a different word than it should be, or their fragments could be discarded, possibly with many false positives.

As it happened for the original bot, every two hours, *Poeta* 2.0 tweets through the account *@poetartificial*, which has about 260 followers.

References

John Charnley, Simon Colton, and Maria Teresa Llano. 2014. The FloWr framework: Automated flowchart construction, optimisation and alteration for creative systems. In *5th International Conference on Computational Creativity, ICCC 2014*, Ljubljana, Slovenia.

Simon Colton, Jacob Goodwin, and Tony Veale. 2012. Full FACE poetry generation. In *Proceedings of 3rd International Conference on Computational Creativity, Dublin, Ireland*, ICCC 2012, pages 95–102, Dublin, Ireland.

Pablo Gervás. 2001. An expert system for the composition of formal Spanish poetry. *Journal of Knowledge-Based Systems*, 14:200–1.

Hugo Gonçalo Oliveira, Tiago Mendes, and Ana Boavida. 2017. Towards finer-grained interaction with a Poetry Generator. In *Proceedings of ProSocrates 2017: Symposium on Problem-solving, Creativity and Spatial Reasoning in Cognitive Systems*, Delmenhorst, Germany, July. CEUR-WS.org.

Hugo Gonçalo Oliveira. 2012. PoeTryMe: a versatile platform for poetry generation. In *Proceedings of ECAI 2012 Workshop on Computational Creativity, Concept Invention, and General Intelligence, Montpellier, France*, C3GI 2012, Montpellier, France, August.

Hugo Gonçalo Oliveira and Ana Oliveira Alves. 2016. Poetry from concept maps – yet another adaptation of PoeTryMe's flexible architecture. In *Proceedings of 7th International Conference on Computational Creativity*, ICCC 2016, Paris, France.

Hugo Gonçalo Oliveira, Diogo Costa, and Alexandre Pinto. 2016. One does not simply produce funny memes! – explorations on the automatic generation of internet humor. In *Proceedings of 7th International Conference on Computational Creativity*, ICCC 2016, Paris, France.

Hugo Gonçalo Oliveira, Raquel Hervás, Alberto Díaz, and Pablo Gervás. 2017. Multilanguage extension and evaluation of a poetry generator. *Natural Language Engineering*, page (in press).

Hugo Gonçalo Oliveira. 2016. Automatic generation of poetry inspired by Twitter trends. In *Knowledge Discovery, Knowledge Engineering and Knowledge Management (Post-conference Proceedings of IC3K — Revised Selected Papers)*, volume 631 of *CCIS*, pages 13–27. Springer.

Hugo Gonçalo Oliveira. 2017a. Comparing and combining Portuguese lexical-semantic knowledge bases. In *Proceedings of the 6th Symposium on Languages, Applications and Technologies (SLATE 2017)*, OASICS, page (in press). Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik.

Hugo Gonçalo Oliveira. 2017b. A survey on intelligent poetry generation: Languages, features, techniques, reutilisation and evaluation. In *Proceedings of 10th International Conference on Natural Language Generation*, INLG 2017, page (in press), Santiago de Compostela, Spain. ACL Press.

Ivan Guerrero, Ben Verhoeven, Francesco Barbieri, Pedro Martins, and Rafael Perez y Perez. 2015. TheRiddlerBot: A next step on the ladder towards creative Twitter bots. In *Proceedings of 6th International Conference on Computational Creativity*, ICCC 2015, pages 315–322, Park City, Utah. Brigham Young University.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Procs. of 14th Conference on Computational Linguistics*, COLING'92, pages 539–545. ACL Press.

Carolyn Lamb, Daniel Brown, and Charles Clarke. 2017. Incorporating novelty, meaning, reaction and craft into computational poetry: a negative experimental result. In *Proceedings of 8th International Conference on Computational Creativity*, ICCC 2017, Atlanta, Georgia, USA.

Hisar Manurung. 2003. *An evolutionary algorithm approach to poetry generation*. Ph.D. thesis, University of Edinburgh.

Joanna Misztal and Bipin Indurkhya. 2014. Poetry generation system with an emotional personality. In *Proceedings of 5th International Conference on Computational Creativity, Ljubljana, Slovenia*, ICCC 2014, Ljubljana, Slovenia, June.

Fam Rashel and Ruli Manurung. 2014. Pemuisi: A constraint satisfaction-based generator of topical indonesian poetry. In *Proceedings of 5th International Conference on Computational Creativity*, ICCC 2014, Ljubljana, Slovenia, June.

Diana Santos and Eckhard Bick. 2000. Providing Internet access to Portuguese corpora: the AC/DC project. In *Proceedings of 2nd International Conference on Language Resources and Evaluation*, LREC 2000, pages 205–210.

Henry Thompson. 1977. Strategy and tactics: a model for language production. In *Papers from the Regional Meeting of the Chicago Linguistic Society*, volume 13, pages 651–668, Chicago, IL, USA. Chicago Linguistic Society.

Berty Chrismartin Lumban Tobing and Ruli Manurung. 2015. A chart generation system for topical metrical poetry. In *Proceedings of the 6th International Conference on Computational Creativity, Park City, Utah, USA*, ICCC 2015, Park City, Utah, USA, Jun.

Jukka M. Toivanen, Oskar Gross, and Hannu Toivonen. 2014. The officer is taller than you, who race yourself! Using document specific word associations in

poetry generation. In *Proceedings of 5th International Conference on Computational Creativity*, ICCC 2014, Ljubljana, Slovenia, June.

Tony Veale, Alessandro Valitutti, and Guofu Li. 2015. Twitter: The best of bot worlds for automated wit. In *Proceedings of 3rd International Conference on Distributed, Ambient, and Pervasive Interactions*, DAPI 2015, pages 689–699, Los Angeles, CA, USA, August.

Tony Veale. 2013. Less rhyme, more reason: Knowledge-based poetry generation with feeling, insight and wit. In *Proceedings of the 4th International Conference on Computational Creativity*, pages 152–159, Sydney, Australia.