Adapting a Generic Platform for Poetry Generation to Produce Spanish Poems

Hugo Gonçalo Oliveira[1], Raquel Hervás[2], Alberto Díaz[2] & Pablo Gervás[2]

hroliv@dei.uc.pt, raquelhb@fdi.ucm.es, albertodiaz@fdi.ucm.es, pgervas@sip.ucm.es

[1]CISUC, University of Coimbra, Portugal
[2]Universidad Complutense de Madrid, Spain

June 11, 2014

Automatic Generation of Poetry

- A number of different methods of techniques applied, e.g.
  - ▶ Template-filling
  - ▶ CBR
  - ▶ Evolutionary
  - ▶ Generate & test
  - ▶ Language model
  - ▶ Constraint programming

Automatic Generation of Poetry

- A number of different methods of techniques applied, e.g.
  - ▶ Template-filling
  - ▶ CBR
  - ▶ Evolutionary
  - ▶ Generate & test
  - ▶ Language model
  - ▶ Constraint programming
- Some dealing with semantics...
  - ▶ Initial prose message / seed words
  - ▶ Semantic networks
  - ▶ Models of semantic similarity

Automatic Generation of Poetry

- A number of different methods of techniques applied, e.g.
  - ► Template-filling
  - ► CBR
  - ► Evolutionary
  - ► Generate & test
  - ► Language model
  - ► Constraint programming
- Some dealing with semantics...
  - ► Initial prose message / seed words
  - ► Semantic networks
  - ► Models of semantic similarity
- Often custom-tailored solutions for specific languages...

Main goal

Explore the effort required for adapting an existing generic platform for poetry generation to a different language

Main goal

Explore the effort required for adapting an existing generic platform for poetry generation to a different language

- PoeTryMe – a generic platform for poetry generation
  - ▶ Highly customizeable
    - ★ Poem structure – stanzas, lines/stanza, syllables/line
    - ★ Semantics – seed words, semantic network
    - ★ Line templates – semantic relation renderings
    - ★ Generation strategy – selection/combination of generated sentences

Main goal

Explore the effort required for adapting an existing generic platform for poetry generation to a different language

- PoeTryMe – a generic platform for poetry generation
  - ▶ Highly customizeable
    - ★ Poem structure – stanzas, lines/stanza, syllables/line
    - ★ Semantics – seed words, semantic network
    - ★ Line templates – semantic relation renderings
    - ★ Generation strategy – selection/combination of generated sentences
  - ▶ Until this collaboration, applied only for **Portuguese**...

Main goal

Explore the effort required for adapting an existing generic platform for poetry generation to a different language

- PoeTryMe – a generic platform for poetry generation
  - ▶ Highly customizeable
    - ★ Poem structure – stanzas, lines/stanza, syllables/line
    - ★ Semantics – seed words, semantic network
    - ★ Line templates – semantic relation renderings
    - ★ Generation strategy – selection/combination of generated sentences
  - ▶ Until this collaboration, applied only for **Portuguese**...
- Could we adapt PoeTryMe to generate Spanish poetry?
 - ▶ What would the **requirements** be?

Main goal

Explore the effort required for adapting an existing generic platform for poetry generation to a different language

- PoeTryMe – a generic platform for poetry generation
  - ▶ Highly customizeable
    - ★ Poem structure – stanzas, lines/stanza, syllables/line
    - ★ Semantics – seed words, semantic network
    - ★ Line templates – semantic relation renderings
    - ★ Generation strategy – selection/combination of generated sentences
  - ▶ Until this collaboration, applied only for **Portuguese**...
- Could we adapt PoeTryMe to generate Spanish poetry?
  - ▶ What would the **requirements** be?
  - ▶ Is its architecture **flexible** enough?
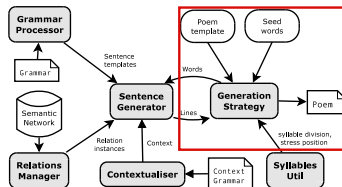 - ▶ How long would it take?

PoeTryMe
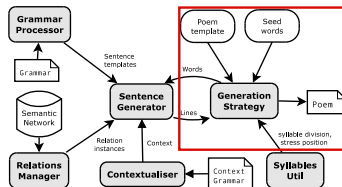## Architecture
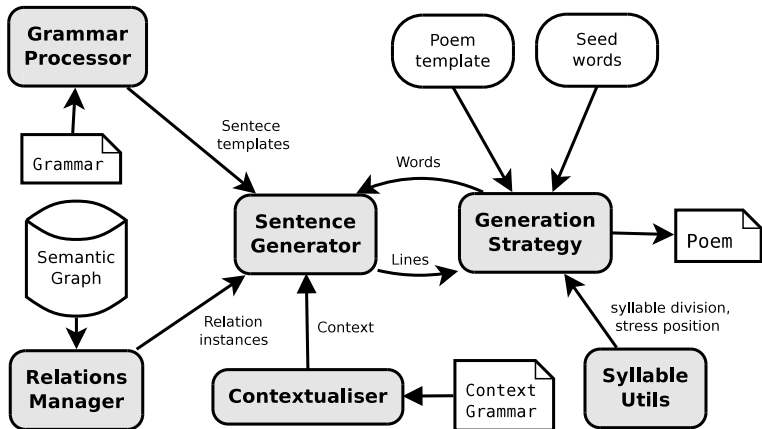
PoeTryMe
Generation Strategy

- Organises sentences according to some heuristics, such that they suit, as much as possible, a target **poem structure** and exhibit **poetic features**

PoeTryMe
Generation Strategy

- Organises sentences according to some heuristics, such that they suit, as much as possible, a target **poem structure** and exhibit **poetic features**
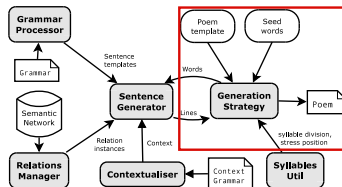


```
#haiku
stanza{line(5);line(7);line(5)}

#sonnet
stanza{line(10:A);line(10:B);line(10:B);line(10:A)}
stanza{line(10:A);line(10:B);line(10:B);line(10:A)}
stanza{line(10:C);line(10:D);line(10:C)}
stanza{line(10:D);line(10:C);line(10:D)}
```

PoeTryMe
Generation Strategy

- Organises sentences according to some heuristics, such that they suit, as much as possible, a target **poem structure** and exhibit **poetic features**

```
#haiku
stanza{line(5);line(7);line(5)}

#sonnet
stanza{line(10:A);line(10:B);line(10:B);line(10:A)}
stanza{line(10:A);line(10:B);line(10:B);line(10:A)}
stanza{line(10:C);line(10:D);line(10:C)}
stanza{line(10:D);line(10:C);line(10:D)}
```
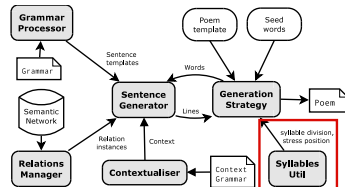
- In this work, **generate & test**
  - Successive generation of sentences for each line
  - Keep the best scoring (metre, rhymes)

PoeTryMe
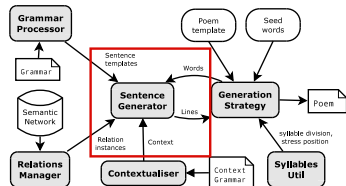## Syllable Utils

- Interface for syllable-related operations
  - **Stress** identification
  - Syllable **division**
  - Termination extraction (for **rhymes**)

PoeTryMe
Sentence Generator

- Generates **syntactically correct sentences** on demand, with the help of...
  - Relations Manager
  - Grammar Processor
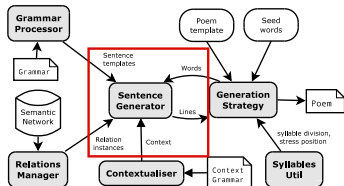
PoeTryMe
Sentence Generator

- Generates **syntactically correct sentences** on demand, with the help of...
  - ▶ Relations Manager
  - ▶ Grammar Processor

- Generation procedure
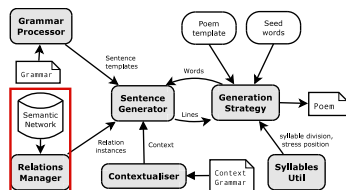 1. Select a random **relation instance** from the Relations Manager
  2. From the Grammar Processor, retrieve a random rule for the relation **predicate**
  3. Insert the relation **arguments** in the rule body
  4. Return the resulting **sentence**

PoeTryMe
Relations Manager

- Manages a **semantic network** that connects words according to their meaning
  - *triple* = (*arg*₁, *predicate*, *arg*₂)

PoeTryMe

Relations Manager
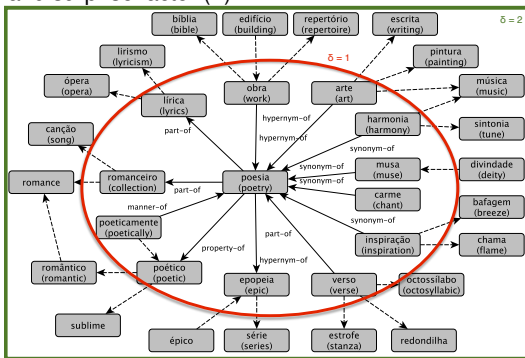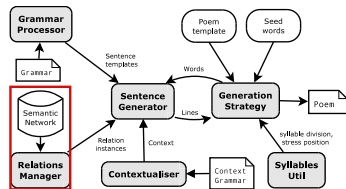
- Manages a **semantic network** that connects words according to their meaning
  - $triple = (arg_1, predicate, arg_2)$
- Selects subgraphs according to
  - Set of seed words
  - Depth (δ) and surprise factor (ν)

PoeTryMe
Grammar Processor

- Reads grammars with **textual renderings** of grammatical sentences that express **semantic relations**
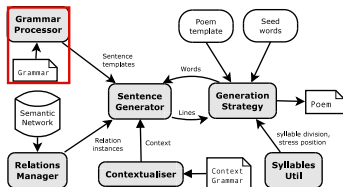
PoeTryMe
Grammar Processor

- Reads grammars with **textual renderings** of grammatical sentences that express **semantic relations**
- Rule names must match relation names in the semantic network


```
HYPERNYM-OF → a <arg1> like a <arg2>
HYPERNYM-OF → <arg2> is a delicious <arg1>
HYPERNYM-OF → <arg2> before <arg1>
```

PoeTryMe
Grammar Processor

- Reads grammars with **textual renderings** of grammatical sentences that express **semantic relations**
- Rule names must match relation names in the semantic network
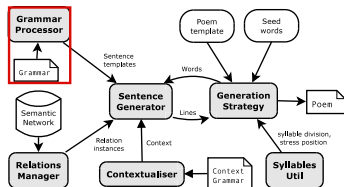


```
HYPERNYM-OF → a <arg1> like a <arg2>
HYPERNYM-OF → <arg2> is a delicious <arg1>
HYPERNYM-OF → <arg2> before <arg1>
```
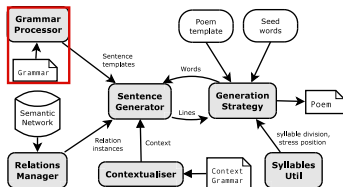
- *a tool like a hammer*
- *mango is a delicious fruit*
- *man before animal*
- *...*

PoeTryMe

Contextualizer

- Keeps track of the selected relations
- And their connection to the domain words
  - ▶ Explains what is *behind* the poem's creation
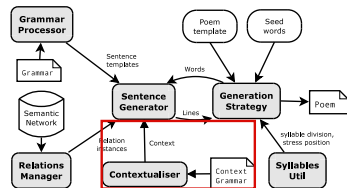
PoeTryMe
Contextualizer

- Keeps track of the selected relations
- And their connection to the domain words
  - ▶ Explains what is *behind* the poem's creation
- Not so explored in this work...
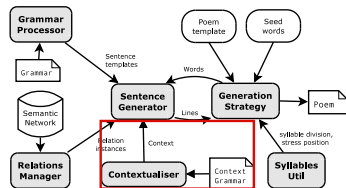
The effort

- What are the requirements to adapt PoeTryMe to Spanish?

The effort

- What are the requirements to adapt PoeTryMe to Spanish?
  - ▶ Spanish semantic network
  - ▶ Spanish semantic relation textual renderings
  - ▶ Spanish syllable utils

The effort

- What are the requirements to adapt PoeTryMe to Spanish?
  - ▶ Spanish semantic network
  - ▶ Spanish semantic relation textual renderings
  - ▶ Spanish syllable utils
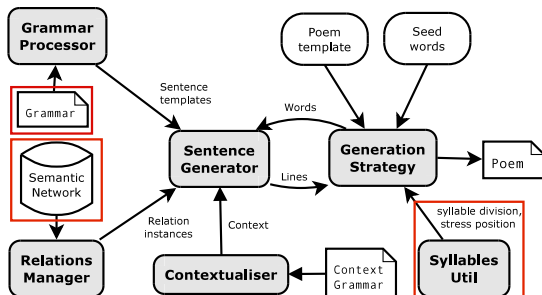
Spanish semantic network

- Spanish Wordnet from the Multilingual Central Repository (MCR)

Spanish semantic network

- Spanish Wordnet from the Multilingual Central Repository (MCR)
- Conversion of synset relations into word relations

Synset relation	{*automóvil, carro, coche*} hypernym-of {*coche_deportivo, deportivo*}
Word triplets	*automóvil* hypernym-of *coche_deportivo* *automóvil* hypernym-of *deportivo* *carro* hypernym-of *coche_deportivo* *carro* hypernym-of *deportivo* *coche* hypernym-of *coche_deportivo* *coche* hypernym-of *deportivo*

Spanish semantic network

- Spanish Wordnet from the Multilingual Central Repository (MCR)
- Conversion of synset relations into word relations

Synset relation	{*automóvil, carro, coche*} hypernym-of {*coche_deportivo, deportivo*}
Word triplets	*automóvil* hypernym-of *coche_deportivo* *automóvil* hypernym-of *deportivo* *carro* hypernym-of *coche_deportivo* *carro* hypernym-of *deportivo* *coche* hypernym-of *coche_deportivo* *coche* hypernym-of *deportivo*

- About 103,000 relation triples between lemmas

Spanish semantic network

- Spanish Wordnet from the Multilingual Central Repository (MCR)
- Conversion of synset relations into word relations

Synset relation	{*automóvil, carro, coche*} hypernym-of {*coche_deportivo, deportivo*}
Word triplets	*automóvil* hypernym-of *coche_deportivo* *automóvil* hypernym-of *deportivo* *carro* hypernym-of *coche_deportivo* *carro* hypernym-of *deportivo* *coche* hypernym-of *coche_deportivo* *coche* hypernym-of *deportivo*

- About 103,000 relation triples between lemmas
- FreeLing Spanish dictionary to handle morphology
  - ▶ 231,296 triples between inflected words

Spanish semantic relation renderings

- Learned automatically from a collection of Spanish human-created poetry
  - ▶ Anthology of Spanish poetry on the Web
  - ▶ WASP knowledge base

Spanish semantic relation renderings

- Learned automatically from a collection of Spanish human-created poetry
  - Anthology of Spanish poetry on the Web
  - WASP knowledge base
- Acquisition procedure
  1. Use the semantic network
  2. Search for lines where two related words co-occur
  3. Extract a line template and generalize it to all relations of the same type

Spanish semantic relation renderings

- Learned automatically from a collection of Spanish human-created poetry
  - ▶ Anthology of Spanish poetry on the Web
  - ▶ WASP knowledge base
- Acquisition procedure
  1. Use the semantic network
  2. Search for lines where two related words co-occur
  3. Extract a line template and generalize it to all relations of the same type
     - ★ Replace related words by `<arg1>` and `<arg2>`

Original line	*El <u>ancho</u> campo me parece <u>estrecho</u>*
	(the wide field looks narrow to me)
Existing relation	*ancho* antonym-of *estrecho*
	(wide antonym-of narrow)
New template	`ANTONYM-OF` → El `<arg1>` campo me parece `<arg2>`

Spanish syllable utils

- Existing modules from WASP

Spanish syllable utils

- Existing modules from WASP
- Implementation of the Syllable Utils interface
  - Syllable division
  - Stress identification
  - Termination extraction

Experimentation settings

- Testing the new instance of PoeTryMe...

Experimentation settings

- Testing the new instance of PoeTryMe...
- Relation renderings learned from
  1. The whole collection of 395 poems – GR+
     - ⋆ 1,285 grammar rules
  2. A subset of the previous, with 64 poems – GR-
     - ⋆ 245 grammar rules

Experimentation settings

- Testing the new instance of PoeTryMe...
- Relation renderings learned from
  1. The whole collection of 395 poems – GR+
     - ★ 1,285 grammar rules
  2. A subset of the previous, with 64 poems – GR-
     - ★ 245 grammar rules
- Semantic relations from
  1. The full set from MCR – SR+
     - ★ 231,296 triples
  2. A subset of SR+ with only synonymy relations – SR-Syn
     - ★ 55,300 triples
  3. A subset of SR+ with only hypernymy relations – SR-Hyp
     - ★ 130,669 triples

Experimentation settings (cont.)

- Target structure: sonnet (no predefined rhyme pattern)

Experimentation settings (cont.)

- Target structure: sonnet (no predefined rhyme pattern)
- Generate & test
  - ▶ Maximum 1000 generations/line
  - ▶ Each additional syllable to the target: +1
  - ▶ Rhymes: −2

Experimentation settings (cont.)

- Target structure: sonnet (no predefined rhyme pattern)
- Generate & test
  - Maximum 1000 generations/line
  - Each additional syllable to the target: +1
  - Rhymes: -2
- Semantic network
  - 100 poems for each depth: $\delta = 1$ and $\delta = 2$
  - $\nu = 0.1$

Experimentation settings (cont.)

- Target structure: sonnet (no predefined rhyme pattern)
- Generate & test
  - Maximum 1000 generations/line
  - Each additional syllable to the target: +1
  - Rhymes: -2
- Semantic network
  - 100 poems for each depth: $\delta = 1$ and $\delta = 2$
  - $\nu = 0.1$
- Seed words
  - *amor* (love)
  - *muerte* (death)
  - *suerte* (luck)
  - *vivir* (to live)
  - *sentir* (to feel)
  - *morir* (to die)

Amount of renderings, relations, depth

δ	GR	SR	% of SR	Evaluation		
				Avg.	**Worst**	**Best**
1	GR−	SR+	0.67%	-8.76	-2	-14
1	GR+	SR+	0.77%	-5.19	0	-10
2	GR−	SR+	13.80%	-8.19	-3	-13
2	GR+	SR+	**17.78%**	-5.93	-1	-12
1	GR−	SR−Hyp	0.56%	**-10.86**	-6	-19
1	GR+	SR−Hyp	0.61%	-4.68	-1	-9
2	GR−	SR−Hyp	13.04%	**-12.03**	-7	-19
2	GR+	SR−Hyp	**15.30%**	-5.53	-1	-10
1	GR−	SR−Syn	0.56%	-6.77	-3	-11
1	GR+	SR−Syn	0.55%	-4.62	0	-9
2	GR−	SR−Syn	2.49%	-8.91	-5	-14
2	GR+	SR−Syn	2.49%	-4.69	0	-10

Best scoring poem

Spanish poem:

> *mi hospedar no quiere albergar*
> *mi pensar no quiere relacionar*
> *mi olvidar no quiere arrojar*
> *mi morir no quiere soportar*
>
> *mi ocupar no quiere trabajar*
> *mi indicar no quiere informar*
> *mi recibir no quiere saludar*
> *mi tragarse no quiere soportar*
>
> *mi albergar no quiere albergar*
> *mi resolver no quiere terminar*
> *mi ocupar no quiere trabajar*
>
> *mi residir no quiere habitar*
> *mi percibir no quiere observar*
> *mi olvidar no quiere descartar*

(rough translation)

> *my hosting wants no holding*
> *my thinking wants no relating*
> *my forgetting wants no throwing*
> *my dying wants no tolerating*
>
> *my busying wants no working*
> *my indicating wants no informing*
> *my receiving wants no greeting*
> *my swallowing wants no tolerating*
>
> *my holding wants no holding*
> *my resolving wants no ending*
> *my busying wants no working*
>
> *my residing wants no living*
> *my perceiving wants no observing*
> *my forgetting wants no discarging*

- Renderings: `GR-`, `SR-Hyp`
- Score: -19
- Not very interesting...

More balanced poem

Spanish poem:

> sordos a las estimas y afectas
> en el dulce amor ejercitados
> en los presentes trabajos y cuidados
> hinchen de tristes desgracias el viento
>
> llamar oler sentir les aprovecha
> y cálidos indómitos cordiales
> por los odiosos los amables males
> hinchen de tristes desgracias el viento
>
> ocupará los actos y la pérdida
> hinchen de tristes desgracias el viento
> que ni la matanza ni el violento
>
> duras puentes romper cual tiernas cañas
> mi lamentar no quiere lamentarse
> mi ocupar no quiere esforzarse

(rough translation)

> deaf to appreciations and affections
> in sweet love exercised
> in present works and cares
> swell the wind with disgrace
>
> calling, smelling, feeling profits them
> and warm cordial untamed
> by the hated, the kind evils
> swell the wind with disgrace
>
> it will fill actions and loss
> well the wind with disgrace
> that neither killing nor violent
>
> hard bridges to break like tender reeds
> my regret does not to want to regret
> my labor does not want to exert

- Renderings: GR+, SR+
- Score: -7

Grammar Rules

	Distinct	**Repetitions**		**Rules**
	renderings	**average**	**maximum**	**from GR**
GR−SR+	57	15.72	259	14.29%
GR+SR+	257	6.83	114	16.31%

- The average number of repetitions per rule used is relatively high
- Only 15% of the grammar rules are used in the generated poems
- Most frequent pattern:
  - ▶ V_HYPERNYM_OF_V → mi <arg2> no quiere <arg1>
    (my <arg2> does not want to <arg1>)

Seed words

- **Experiment 1:** in the original poem collection
  - Most frequent terms (+Freq): *yo* (I), *gente* (people), *tierra* (dust), *amor* (love), *vida* (life), and *ser* (to be)
  - Least frequent terms (-Freq): *abismo* (abyss), *austro* (south wind), *tempestades* (storms), *detenerse* (to stop), *creer* (to believe), and *combatir* (to fight)
  - Always with GR+SR+

Seeds	Triples used	Avg score
+Freq	75,622	-5.78
-Freq	9,014	-7

Seed words

- **Experiment 1:** in the original poem collection
  - Most frequent terms (+Freq): *yo* (I), *gente* (people), *tierra* (dust), *amor* (love), *vida* (life), and *ser* (to be)
  - Least frequent terms (-Freq): *abismo* (abyss), *austro* (south wind), *tempestades* (storms), *detenerse* (to stop), *creer* (to believe), and *combatir* (to fight)
  - Always with GR+SR+

Seeds	Triples used	Avg score
+Freq	75,622	-5.78
-Freq	9,014	-7

- **Experiment 2:** use only one seed (*amor*) + PageRank to obtain the top-5 most relevant words
  - Extended seed set: *amor*, *amores*, *cariño*, *afectas*, *afecta*
  - Best scoring poem with GR+SR+

Discussion

- PoeTryMe integrates classic approaches for poetry generation
  - Semantic input $\rightarrow$ semantic-based
  - Grammar of semantic renderings $\rightarrow$ Syntax-aware
  - When renderings are learned from existing poems $\rightarrow$ parallelism with CBR

Discussion

- PoeTryMe integrates classic approaches for poetry generation
  - Semantic input → semantic-based
  - Grammar of semantic renderings → Syntax-aware
  - When renderings are learned from existing poems → parallelism with CBR
- Current limitations
  - Learned patterns must include two related words (small percentage)
  - Sometimes, contiguous lines lack connection
  - Line templates already impose a starting number of syllables
    - ★ Explains the low proportion of used patterns, from those learned

Discussion

- PoeTryMe integrates classic approaches for poetry generation
    - Semantic input → semantic-based
    - Grammar of semantic renderings → Syntax-aware
    - When renderings are learned from existing poems → parallelism with CBR
- Current limitations
    - Learned patterns must include two related words (small percentage)
    - Sometimes, contiguous lines lack connection
    - Line templates already impose a starting number of syllables
        - ★ Explains the low proportion of used patterns, from those learned
- Nevertheless...
    - High degree of variation
    - Metre often satisfied
    - Syntactically and semantically coherent lines
    - Semantic connection between used terms

Concluding remarks

- First effort on the adaptation of PoeTryMe to Spanish

Concluding remarks

- First effort on the adaptation of PoeTryMe to Spanish
  - ▶ Quite straightforward, quite successful
    - ★ There were available Spanish resources that suited this purpose!

Concluding remarks

- First effort on the adaptation of PoeTryMe to Spanish
  - Quite straightforward, quite successful
    - There were available Spanish resources that suited this purpose!
  - Confirmed that PoeTryMe is flexible enough

Concluding remarks

- First effort on the adaptation of PoeTryMe to Spanish
  - ▶ Quite straightforward, quite successful
    - ★ There were available Spanish resources that suited this purpose!
  - ▶ Confirmed that PoeTryMe is flexible enough
- Future:
  - ▶ More tests, exploiting other features of PoeTryMe, e.g.
    - ★ PageRank
    - ★ Contextualizer
    - ★ Sentiment
    - ★ Learning renderings from non-poetical text

Concluding remarks

- First effort on the adaptation of PoeTryMe to Spanish
  - ▶ Quite straightforward, quite successful
    - ★ There were available Spanish resources that suited this purpose!
  - ▶ Confirmed that PoeTryMe is flexible enough
- Future:
  - ▶ More tests, exploiting other features of PoeTryMe, e.g.
    - ★ PageRank
    - ★ Contextualizer
    - ★ Sentiment
    - ★ Learning renderings from non-poetical text
  - ▶ More evaluation
    - ★ Improve the computation of automatic scores
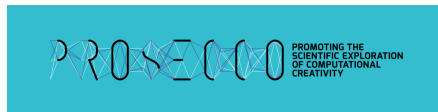   - ★ Evaluation of the semantics?
    - ★ Manual?

Concluding remarks

- First effort on the adaptation of PoeTryMe to Spanish
  - ▶ Quite straightforward, quite successful
    - ⋆ There were available Spanish resources that suited this purpose!
  - ▶ Confirmed that PoeTryMe is flexible enough
- Future:
  - ▶ More tests, exploiting other features of PoeTryMe, e.g.
    - ⋆ PageRank
    - ⋆ Contextualizer
    - ⋆ Sentiment
    - ⋆ Learning renderings from non-poetical text
  - ▶ More evaluation
    - ⋆ Improve the computation of automatic scores
    - ⋆ Evaluation of the semantics?
    - ⋆ Manual?
  - ▶ Adaptation to other languages – English?
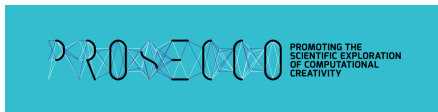
Thank you

Acknowledgements:

CONCRETE

Thank you

Questions?

Acknowledgements:

CONCRETE