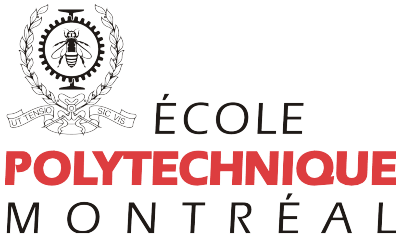
A preprocessing system to include imaginative animations according to text in educational applications.



Plan

The GITAN Project

Grammar for Interpretation of Text and ANimations

Proposing a universal model to manage transition from a textual content to a graphical animated representation.

The GITAN Project

Grammar for Interpretation of Text and ANimations

Proposing a universal model to manage transition from a textual content to a graphical animated representation.

A long-term project

Began at the end of 2009. First prototype planned at the end of 2010.

Case studied in this communication

A system dedicated to build a language learning software application.

Case studied in this communication

A system dedicated to build a language learning software application.

- An immediate engineering problem to solve: how to display imaginative content ?

Case studied in this communication

A system dedicated to build a language learning software application.

- An immediate engineering problem to solve: how to display imaginative content ?
- An opportunity to investigate actual limitations of text to animation propositions.

Case studied in this communication

A system dedicated to build a language learning software application.

- An immediate engineering problem to solve: how to display imaginative content ?
- An opportunity to investigate actual limitations of text to animation propositions.
- An investigation on limitation of text to graphics engines regarding imaginative sentences

Case studied in this communication

A system dedicated to build a language learning software application.

- An immediate engineering problem to solve: how to display imaginative content ?
- An opportunity to investigate actual limitations of text to animation propositions.
- An investigation on limitation of text to graphics engines regarding imaginative sentences

Difficult aspect of this work.

Open text input with restricted bag of words. Any word combination is allowed. Each semantically valid sentence must be displayed.

Application principles

A system dedicated to build a language learning software application.

Targeted Learning software for Text to Animation

A system dedicated to build a language learning software application.

Targeted Learning software for Text to Animation

- A bag of words is proposed to a student

- $\{prince, transforms, into, the, castle, in, his, toad, himself, a\}$

A system dedicated to build a language learning software application.

Targeted Learning software for Text to Animation

- A bag of words is proposed to a student
- The learning software displays an animation representing the targeted sentence.

- $\{prince, transforms, into, the, castle, in, his, toad, himself, a\}$

A system dedicated to build a language learning software application.

Targeted Learning software for Text to Animation

- A bag of words is proposed to a student
- The learning software displays an animation representing the targeted sentence.
- The student composes his sentence and the software display an animation representing this sentence.

- $\{prince, transforms, into, the, castle, in, his, toad, himself, a\}$
- *The prince transforms himself into a toad*

A system dedicated to build a language learning software application.

The student can compare the animation resulting of his own words combination.

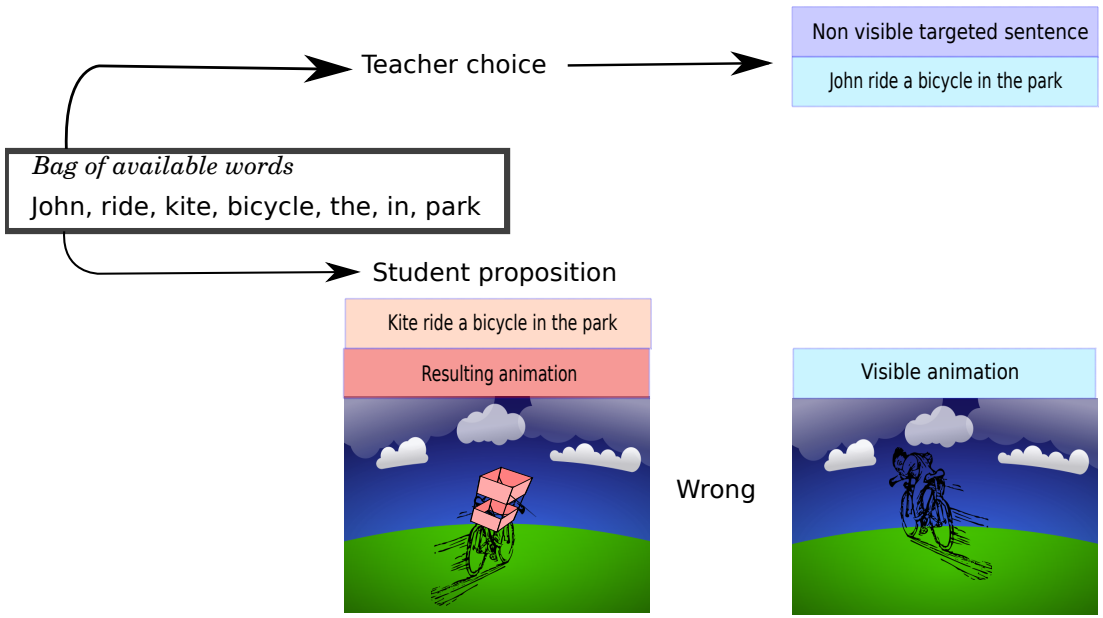


Figure: Synaptic representation of proposed application

A system dedicated to build a language learning software application.

Application may meet situations where the animation does not respect physical laws and common sense.

Example

A system dedicated to build a language learning software application.

Application may meet situations where the animation does not respect physical laws and common sense.

Example

- A bag of words, including the 10 following terms: {**Jack, rides, with, bicycle, park, the, kite, runs, in, his**}.

A system dedicated to build a language learning software application.

Application may meet situations where the animation does not respect physical laws and common sense.

Example

- A bag of words, including the 10 following terms: {**Jack, rides, with, bicycle, park, the, kite, runs, in, his**}.
- *Jack rides his bicycle in the park. The kite runs in the park.*

A system dedicated to build a language learning software application.

Application may meet situations where the animation does not respect physical laws and common sense.

Example

- A bag of words, including the 10 following terms: {**Jack, rides, with, bicycle, park, the, kite, runs, in, his**}.
- *Jack rides his bicycle in the park. The kite runs in the park.*
- Can also be *The bicycle rides Jack. The kite rides the bicycle.*

A system dedicated to build a language learning software application.

Semantic cases

A system dedicated to build a language learning software application.

Semantic cases

- **Position case**: *The kite rides the bicycle*. Can be represented by a graphic engine.

A system dedicated to build a language learning software application.

Semantic cases

- **Position case**: *The kite rides the bicycle*. Can be represented by a graphic engine.
- **Action case**: *The chair eats the cat. The chair eats on the cat*.

A system dedicated to build a language learning software application.

Semantic cases

- **Position case**: *The kite rides the bicycle*. Can be represented by a graphic engine.
- **Action case**: *The chair eats the cat. The chair eats on the cat*.
- **Transformation case**: *The prince transforms himself into a toad*.

Theoretical view and previous work

Limitations of Text to Image systems: theoretical view

- Tversky : "correspondences between mental and graphical representations *suggest cognitive correspondences between mental spaces and real ones*" *Tversky (2002)*.
- Johnson-Laird : consider that there is some mental representation that *cannot be visualized* (*Johnson-Laird (1998)*, page 442)
- Adorni: cognitive transformation should be relevant to a computer AI problem *Adorni 1984* .

Limitations of Text to Image systems: theoretical view

In pictural arts, the correspondences for mental representations permitted by imagination, are obtained by a cognitive transformation of physical law, natural spaces and transgression of common sense to adapt an animation or a static image to the mental representation.

Limitations of Text to Image systems: existing systems

Some samples of existing applications and their limitations

[1] 18 characters, 67 behaviors, and 31 backgrounds

Limitations of Text to Image systems: existing systems

Some samples of existing applications and their limitations

- WordsEye : authors consider that *it is infeasible to fully capture the semantic content of language in graphics*. In *Coyne 2001* page 496.

[1] 18 characters, 67 behaviors, and 31 backgrounds

Limitations of Text to Image systems: existing systems

Some samples of existing applications and their limitations

- WordsEye : authors consider that *it is infeasible to fully capture the semantic content of language in graphics*. In *Coyne 2001* page 496.
- e-Hon : operates in a closed semantic field[1] but uses an AI engine to try to solve most of the semantic cases. This system constrains sentences *using ontological knowledge in real time Sumi 2006*

[1] 18 characters, 67 behaviors, and 31 backgrounds

Limitations of Text to Image systems: existing systems

Some samples of existing applications and their limitations

- WordsEye : authors consider that *it is infeasible to fully capture the semantic content of language in graphics*. In *Coyne 2001* page 496.
- e-Hon : operates in a closed semantic field[1] but uses an AI engine to try to solve most of the semantic cases. This system constrains sentences *using ontological knowledge in real time Sumi 2006*
- Confucius : an ontology of eventive verbs is used to constrain the representation to commonsense. In *Ma 2006* page 109.

[1] 18 characters, 67 behaviors, and 31 backgrounds

Proposed System

A intermediate way to solve the problem of open semantic content representation.

Proposition

A intermediate way to solve the problem of open semantic content representation.

Proposition

- Enumerating all the syntacticaly valid sentences that may potentially be produced for a given bag of words

A intermediate way to solve the problem of open semantic content representation.

Proposition

- Enumerating all the syntacticaly valid sentences that may potentially be produced for a given bag of words
- Clustering of those sentences into groups according to their meaning similarity

A intermediate way to solve the problem of open semantic content representation.

Proposition

- Enumerating all the syntacticaly valid sentences that may potentially be produced for a given bag of words
- Clustering of those sentences into groups according to their meaning similarity
- Pre-producing an unique animation for each cluster of sentences.

Architecture.

A bag of words

cat, eats, on, the, chair, in, his

Architecture.

A bag of words

cat, eats, on, the, chair, in, his

Modules

Architecture.

A bag of words

cat, eats, on, the, chair, in, his

Modules

- Sentence generator (SG)

Architecture.

A bag of words

cat, *eats*, *on*, *the*, *chair*, *in*, *his*

Modules

- Sentence generator (SG)
- Language model Filter (LMF)

Architecture.

A bag of words

cat, *eats*, *on*, *the*, *chair*, *in*, *his*

Modules

- Sentence generator (SG)
- Language model Filter (LMF)
- Clustering algorithm (CA)

Architecture.

Sentence generator (SG)

Built with a limited set of flexible generative grammar rules implemented in Prolog. Rules cover verbal phrases, noun phrases and prepositional phrases and allow the generation of sentences from a bag of words.

$vp(Features, BagIn, BagOut) --> lex(v, Features, BagIn, Bag1), np(, Bag1, Bag2), pp(Bag2, BagOut)$

Architecture.

Language Model Filter (LMF)

A Language model (LM) is trained from a corpus which domain is related to the targeted application. Each candidate sentence proposed by the *Sentence Generator* is filtered by using an estimation of its probability, regarding LM.

Probability $P(w_1, \ldots, w_n)$ to observe a sentence composed of words $w_1 \ldots . w_n$ in the modeled corpus is estimated by the product of probabilities of the individual appearance of words contained in sequence $P(w_{1,n}) \approx P(w_1)P(w_2) \ldots P(w_n)$. A bi-gram model is used to select sentences.

Architecture.

Clustering algorithm (CA)

An analysis of a sentence trough chunking that identifies the constituents (noun phrases, verb phrases, etc.). Considering the list *l* of *n* sentences 1...*n* kept by LMF, we generate a function *f_similarity* for the first sentence *s*1 of *l*. All sentences are evaluated by function, and clustered if accepted. Process is iterated until the list of sentences is empty.

Example of resulting cluster:
[*Jack*/*NC*][*rides*/*VC*][*a bicycle*/*NC*], [*Jack*/*NC*][*runs*/*VC*][*the bicycle*/*NC*],
[*Jack*/*NC*][*rides*/*VC*][*the bicycle*/*NC*]

Evaluating the capacities of the algorithms

Experiment

Evaluating the capacities of the algorithms

Experiment

- 10 bags of 10 words.

Evaluating the capacities of the algorithms

Experiment

- 10 bags of 10 words.
- Bags of words come from exercises included in an English student's book.

Evaluating the capacities of the algorithms

Experiment

- 10 bags of 10 words.
- Bags of words come from exercises included in an English student's book.
- We use 6 and 10 words sets from the bag and apply SG, LMF and CA.

Evaluating the capacities of the algorithms

Experiment

- 10 bags of 10 words.
- Bags of words come from exercises included in an English student's book.
- We use 6 and 10 words sets from the bag and apply SG, LMF and CA.
- We count sentences generated in SG, kept in LMF, and how many clusters remain in CA.

Preliminary experimental results

Words	Generated sentences (SG)	Correct sentences (LMF)	Sentences clusters (CA)
6	25	23	7
10	460	280	20

Discussions:

Preliminary experimental results

Words	Generated sentences (SG)	Correct sentences (LMF)	Sentences clusters (CA)
6	25	23	7
10	460	280	20

Discussions:

- For a 6 words bag, we need to pre-process 7 animations variants corresponding to potential sentences

Preliminary experimental results

Words	Generated sentences (SG)	Correct sentences (LMF)	Sentences clusters (CA)
6	25	23	7
10	460	280	20

Discussions:

- For a 6 words bag, we need to pre-process 7 animations variants corresponding to potential sentences
- For a 10 words bag, we need to pre-process 20 animations variants corresponding to potential sentences

Preliminary experimental results

Words	Generated sentences (SG)	Correct sentences (LMF)	Sentences clusters (CA)
6	25	23	7
10	460	280	20

Discussions:

- For a 6 words bag, we need to pre-process 7 animations variants corresponding to potential sentences
- For a 10 words bag, we need to pre-process 20 animations variants corresponding to potential sentences

Preliminary results are sufficient to build an application prototype.

Next step: a test corpus generator

The system will be used as a component of text-to-animation application who automatically produces test sentences for evaluation purposes.

Targeted application

Next step: a test corpus generator

The system will be used as a component of text-to-animation application who automatically produces test sentences for evaluation purposes.

Targeted application

- Semantic field of a text to animation system is defined by a set of words (verbs, nouns).

Next step: a test corpus generator

The system will be used as a component of text-to-animation application who automatically produces test sentences for evaluation purposes.

Targeted application

- Semantic field of a text to animation system is defined by a set of words (verbs, nouns).
- All possible sentences are generated with this system.

Next step: a test corpus generator

The system will be used as a component of text-to-animation application who automatically produces test sentences for evaluation purposes.

Targeted application

- Semantic field of a text to animation system is defined by a set of words (verbs, nouns).
- All possible sentences are generated with this system.
- Each clusters C of sentence S is associated with a first order predicate P.

Next step: a test corpus generator

The system will be used as a component of text-to-animation application who automatically produces test sentences for evaluation purposes.

Targeted application

- Semantic field of a text to animation system is defined by a set of words (verbs, nouns).
- All possible sentences are generated with this system.
- Each clusters C of sentence S is associated with a first order predicate P.
- S is used to validate Semantic parser regarding P.

Next step: a test corpus generator

The system will be used as a component of text-to-animation application who automatically produces test sentences for evaluation purposes.

Targeted application

- Semantic field of a text to animation system is defined by a set of words (verbs, nouns).
- All possible sentences are generated with this system.
- Each clusters C of sentence S is associated with a first order predicate P.
- S is used to validate Semantic parser regarding P.
- P is used to validate representation capacities of graphic engine regarding any C.

Conclusions

We presented an original component to support text to animation applications in the context of imaginative sentences.

Conclusions

Conclusions

We presented an original component to support text to animation applications in the context of imaginative sentences.

Conclusions

- An attempt to evaluate the capacities of a system to elaborate imaginative-like text to animation system.

Conclusions

We presented an original component to support text to animation applications in the context of imaginative sentences.

Conclusions

- An attempt to evaluate the capacities of a system to elaborate imaginative-like text to animation system.
- An intermediate solution to build application of Text to Image without semantic content restriction.

Conclusions

We presented an original component to support text to animation applications in the context of imaginative sentences.

Conclusions

- An attempt to evaluate the capacities of a system to elaborate imaginative-like text to animation system.
- An intermediate solution to build application of Text to Image without semantic content restriction.
- A way to test capacities of Text to Image application for difficult semantic meaning of open text.

Questions.