Analysis and evaluation of interactive stories

From SimDate3D to DoPPioGioco

Author: Matteo Silvestro

Supervisors: Vincenzo Lombardo
Roberto Esposito

Università degli Studi di Torino

September 13, 2017

Table of Contents

Interactive storytelling
What is interactive storytelling

- *Interactive storytelling* (IS) is a form of digital entertainment in which the storyline is not predetermined.
- The author creates the setting, characters, and situation which the narrative must address, but the player experiences a unique story based on their interactions with the story world.

Interactive storytelling
What is interactive storytelling

- *Interactive storytelling* (IS) is a form of digital entertainment in which the storyline is not predetermined.
- The author creates the setting, characters, and situation which the narrative must address, but the player experiences a unique story based on their interactions with the story world.
- Popular commercial IS games: visual novels (*CLANNAD*, *Steins;Gate*), TellTale games (*The Walking Dead*).

Interactive storytelling
What is interactive storytelling

- *Interactive storytelling* (IS) is a form of digital entertainment in which the storyline is not predetermined.
- The author creates the setting, characters, and situation which the narrative must address, but the player experiences a unique story based on their interactions with the story world.
- Popular commercial IS games: visual novels (*CLANNAD*, *Steins;Gate*), TellTale games (*The Walking Dead*).
- Two main types:
  1. *character-based*, in which story emerges from interaction between characters;
  2. *story-based*, in which the author designs a *story graph*, that is the collection of static story units and their connections.

Example of character-based IS system 1

SimDate3D level one

SimDate3D level one (SD1), where the goal is to achieve that a couple - Thomas and Barbara - gets to the cinema. Three possible endings

1. characters get to the cinema safely;
2. characters get angry and part;
3. characters interaction is too positive, so they decide to skip the cinema and head home.

Example of character-based IS system 2

SimDate3D level two

SimDate3D level two (SD2), where Thomas is dating two girls (Barbara and Nataly) at the same time. Four possible endings

1. Thomas stays with Barbara and breaks up with Nataly;
2. Thomas stays with Nataly and breaks up with Barbara;
3. both girls break up with Thomas
4. Thomas stays in the relationship with both girls.

Example of story-based IS system

DoppioGioco

DoppioGioco is an interactive storytelling system in which the player reacts to the audience emotional response to a unit. This way a linear story arises from the story graph.

Introduction
The problem we want to address

- *Problem*: How to evaluate IS systems?
- A major challenge in designing *storytelling systems* is the evaluation of the resulting narrative.
- As of now, while there are many ways to analyze linear stories, there has not been many researches into the field of interactive stories $\rightarrow$ interactive storytellers have no assistance in managing the narrative, that can become quite complex.

How we can address the problem

User surveys

First approach: User surveys.

- *Pros*: If well designed, capture the quality correctly.
- *Cons*: Costly, time consuming (possible combinatorial explosion of story space).

On the right, *Engagement Sample Questionnaire*[1].

ESQ Part One: Demographics (gender, age, frequency and amount of playing, favourite game / genre etc.)
ESQ Part Two: Before the experience
Q1. Please indicate below the extent to which you agree or disagree with this sentence: "I want to begin the experience" (to quantify the users Continuation Desire (CD))

Disagree strongly	Disagree moderately	Disagree a little	Neither agree nor disagree	Agree a little	Agree moderately	Agree strongly	Other

Q2. "What makes you want/not want to begin?" (to identify the user's CD and objective)
ESQ Part Tree: During the experience
Q3. Please indicate the extent to which you agree or disagree with this sentence: "I want to continue the experience now!" (Response options as in Q1)
Q4. "What makes you want/not want to continue and why?" (to identify the source of the user's CD, objective and intended accomplishment)
Q5. "What do you feel now?" (to indicate the user's affect)
Q6. "What do you want to do next?" (to identify the user's activity and objective)
ESQ Part Four: After the experience
Q7. Please indicate the extent to which you agree or disagree with this sentence: "I want to try again" (to identify the source of the user's CD, objective and accomplishment) – (Response options as in Q1)
Q8, "What makes you want/not want to try again (in the application / experience) and why?"
Q9. "What do you feel now?" (to indicate the user's affect)

[1] Henrik Schoenau-Fog, *Hooked! - Player Engagement in Digital Games as a Driver in Interactive Storytelling, Learning Scenarios and Teaching*

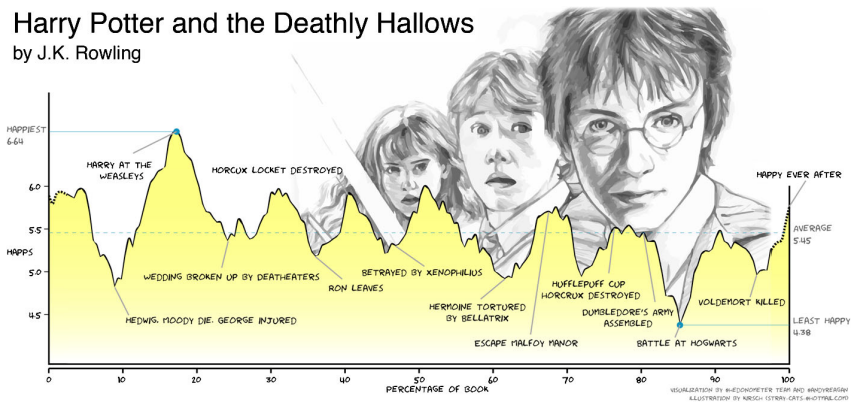
How we can address the problem

Technical evaluations

Second approach: Technical evaluations, such as comparing length and complexity of generated stories.

- *Pros*: Cheap, feasible for large scale story space.
- *Cons*: Loosely related to enjoyment of stories by the user.

On the right, *emotional arc*.[2]



[1] Andrew J. Reagan et al., *The emotional arcs of stories are dominated by six basic shapes*

How we can address the problem

Computer-assisted evaluation

Proposed approach: Computer-assisted evaluation.
Put together the best of both worlds!

Computer-assisted evaluation
Two proposals

- *A 2013 paper proposal*: In the paper *Towards Automatic Story Clustering for Interactive Narrative Authoring* by Michal Bída et al. a simple solution was proposed.
  1. Tailored for *character-based* IS systems.
  2. Based on aggregation (clustering) of linear stories extracted from the IS system.
  3. Evaluation of clusters is very system-specific (related to story endings).
  4. Relies on story designer opinion.

Computer-assisted evaluation
Two proposals

- *A 2013 paper proposal*: In the paper *Towards Automatic Story Clustering for Interactive Narrative Authoring* by Michal Bída et al. a simple solution was proposed.
  1. Tailored for *character-based* IS systems.
  2. Based on aggregation (clustering) of linear stories extracted from the IS system.
  3. Evaluation of clusters is very system-specific (related to story endings).
  4. Relies on story designer opinion.
- *Our proposal*: Expanding the previous approach.
  1. Tailored for *story-based* IS systems.
  2. Based on aggregation (clustering) of linear stories extracted from the IS system.
  3. Evaluation of clusters is based on quality scores and hence, in principle, more general.
  4. Relies on user quality scores and story designer opinion.

Table of Contents

Methodology

Brief overview of the semi-automatic narrative analysis

How it works

1. the story designer uses his system to generate large number of linear stories;

Methodology

Brief overview of the semi-automatic narrative analysis

How it works

1. the story designer uses his system to generate large number of linear stories;
2. the story designer runs the clustering algorithm dividing the stories into groups;

Methodology

Brief overview of the semi-automatic narrative analysis

How it works

1. the story designer uses his system to generate large number of linear stories;
2. the story designer runs the clustering algorithm dividing the stories into groups;
3. the story designer now needs to see only several stories from each group to evaluate the system.

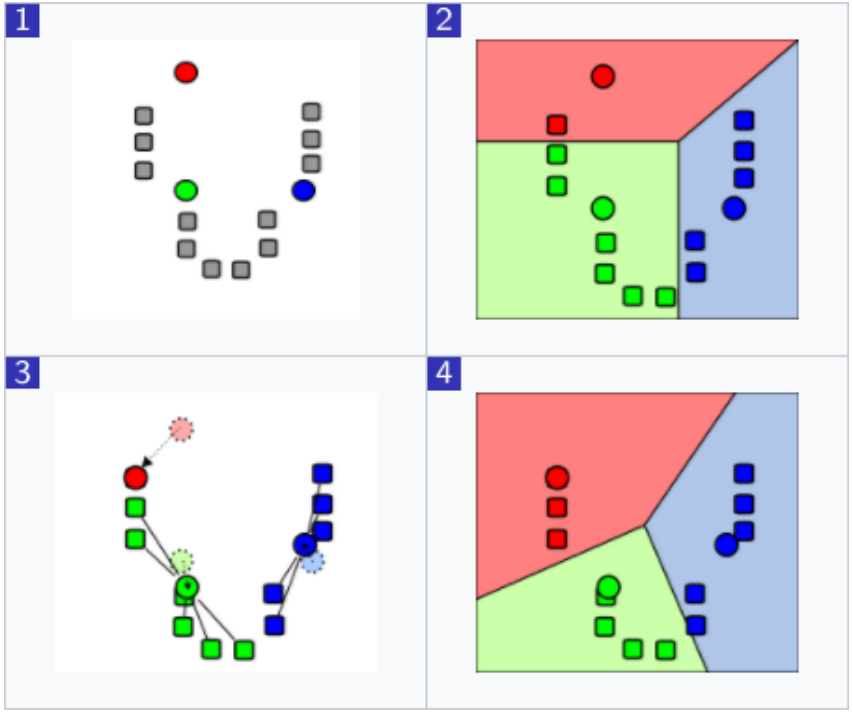
Generating linear stories
IS systems analyzed

SimDate3D level one (SD1)

SimDate3D level two (SD2)

Clustering
k-means in a nutshell

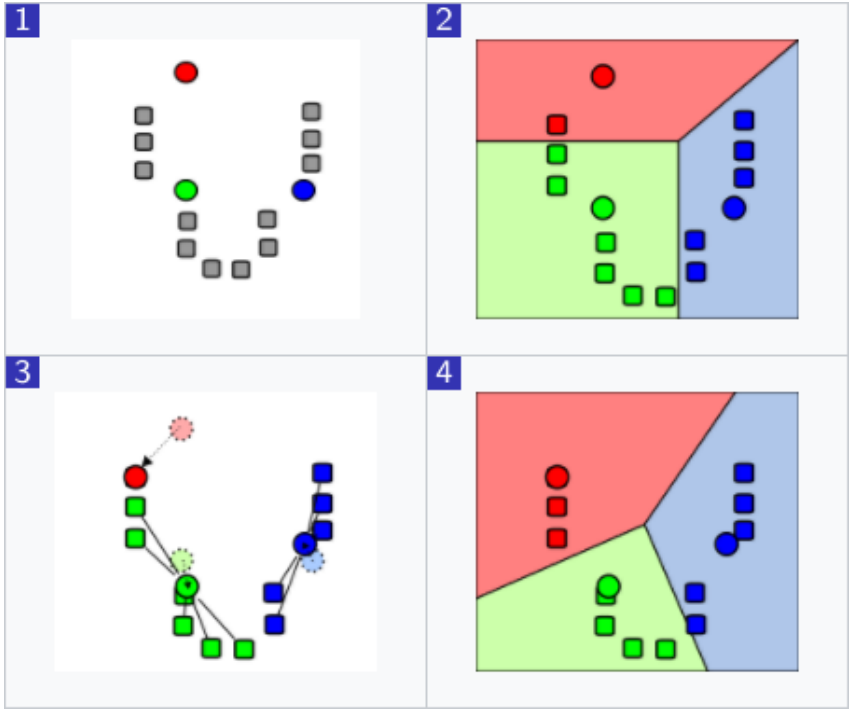
k-means partitions n samples (stories) into k clusters (group of stories) such that the distance between samples within a single cluster is minimized.



Clustering
k-means in a nutshell

k-means algorithm

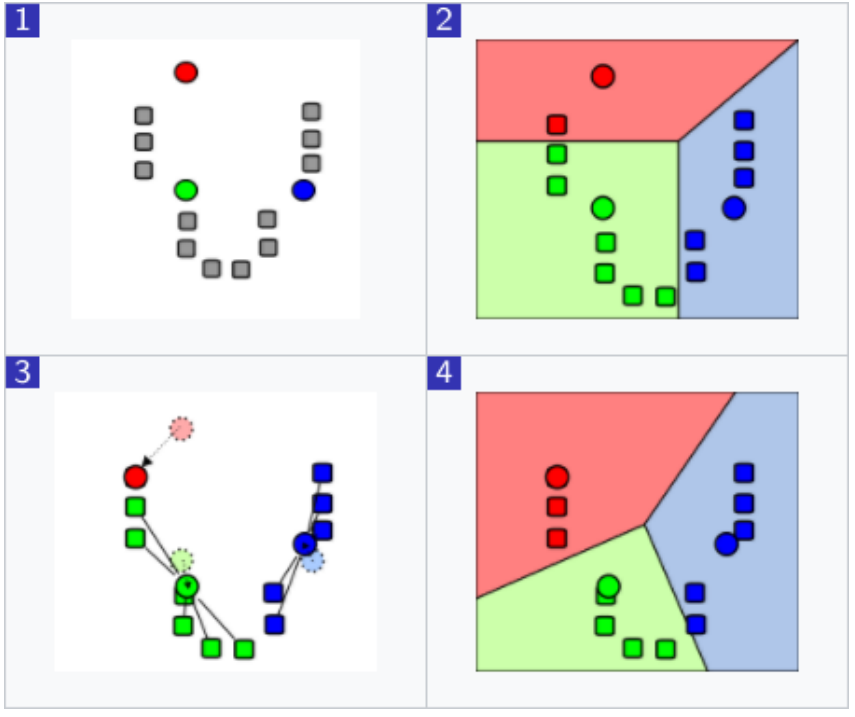
1. initialize the starting position of clusters;



Clustering
k-means in a nutshell

k-means algorithm

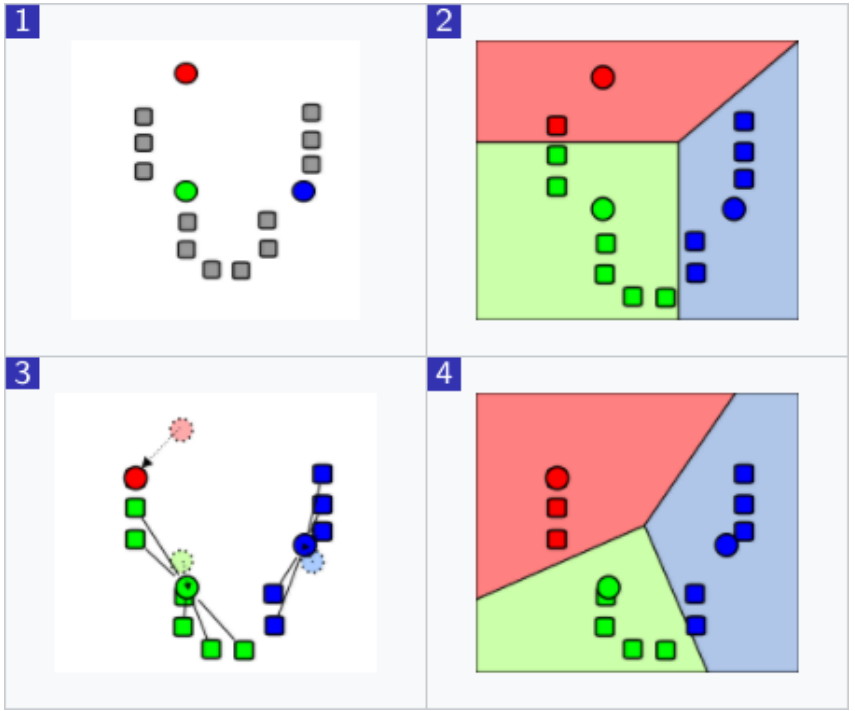
1. initialize the starting position of clusters;
2. assign the nearest cluster to each sample;



Clustering
k-means in a nutshell

k-means algorithm

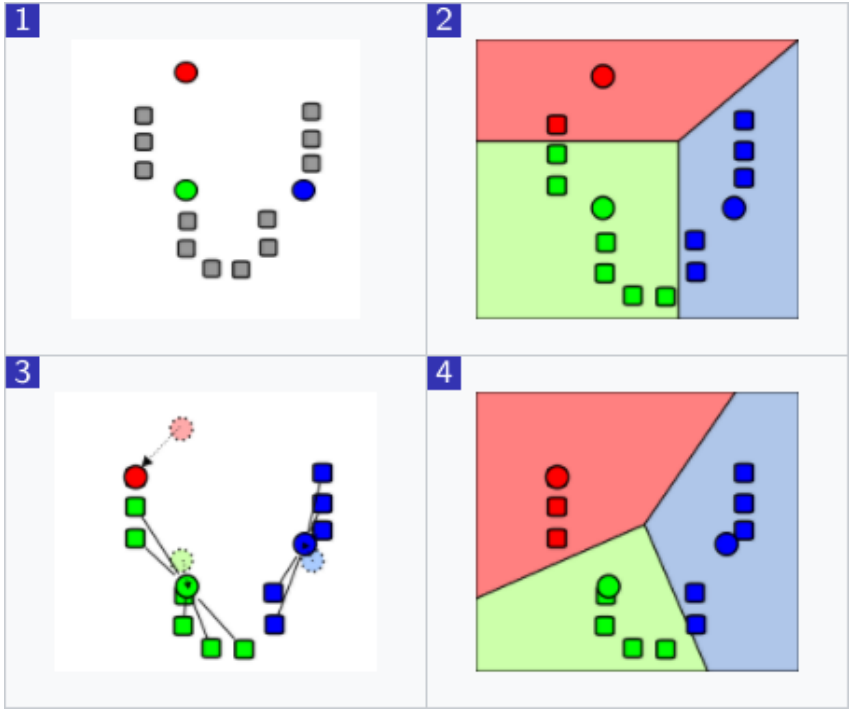
1. initialize the starting position of clusters;
2. assign the nearest cluster to each sample;
3. for each cluster, compute the average of members in the cluster and shift the cluster towards this new average;



Clustering
k-means in a nutshell

k-means algorithm

1. initialize the starting position of clusters;
2. assign the nearest cluster to each sample;
3. for each cluster, compute the average of members in the cluster and shift the cluster towards this new average;
4. repeat the last two steps until the clusters become stable or a maximum number of steps is reached.



Distances

How we should choose distances

- If we define a good "distance" between stories, then each cluster will contain stories that are similar to each other. In this case four distances were tested
  - three based on the action sequences;
  - one based on the tension curves.
- The only distance that we will use in our proposal is the tension curves based one.

Action sequences
Distances for action sequences

- To compute the distance between *action sequences*, first encode the action sequences as action strings in this way
  1. in SD1 each action type is encoded as a letter (upper and lower cases letters and numbers, around 50 possible actions);
  2. is SD2 one more letter per action representing the character performing the action.

Action sequences
Distances for action sequences

- To compute the distance between *action sequences*, first encode the action sequences as action strings in this way
  1. in SD1 each action type is encoded as a letter (upper and lower cases letters and numbers, around 50 possible actions);
  2. is SD2 one more letter per action representing the character performing the action.
- SD1 action sequence: "AABDBHHMNIOIODFGLLCCPHPH-PHPDMHNUUUUUVQWDP".

Action sequences
Distances for action sequences

- To compute the distance between *action sequences*, first encode the action sequences as action strings in this way
  1. in SD1 each action type is encoded as a letter (upper and lower cases letters and numbers, around 50 possible actions);
  2. is SD2 one more letter per action representing the character performing the action.
- SD1 action sequence: "AABDBHHMNIOIODFGLLCCPHPH-PHPDMHNUUUUUVQWDP".
- SD2 action sequence: "ABACDBDEF4APDQDNAOD1AkDE-DrACA9D5DQDNAOAPACDEA2D2A2DQDNAPAOD1ACAK-DKALDQDNAPDMACANDQDNAPAOACAPDQDNAOFQDN-ACAJAPDQDNAOFQDQF4D1". (D – Thomas, A – Barbara and F – Nataly, e.g. "AB" marks Barbara performing action "set focus".)

Action sequences

More on metrics for action strings

- Three standard string difference metrics tested
  1. *Levenshtein distance*;
  2. *Jaccard index*;
  3. *Jaro-Winkler distance*.

Action sequences

More on metrics for action strings

- Three standard string difference metrics tested
  1. *Levenshtein distance*;
  2. *Jaccard index*;
  3. *Jaro-Winkler distance*.
- We will not go into the details for *Jaro-Winkler distance* and *Levenshtein distance*, because it is quite complex to show how works.
- As an example let us see the distance for *Jaccard index* between
  1. "FLOW" and "LAWN";
  2. "STRING" and "GNIRTS".

Action sequences
Jaccard index

- The Jaccard index is simply defined as the ratio between the number of common letters and the total number of letters.
- "FLOW" and "LAWN" have
  - "L" and "W" in common (2);
  - the unique letters "F", "L", "O", "W", "A", "N" (6);

  hence the Jaccard index is $\frac{2}{6} \approx 33\%$.

Action sequences
Jaccard index

- The Jaccard index is simply defined as the ratio between the number of common letters and the total number of letters.
- "FLOW" and "LAWN" have
  - "L" and "W" in common (2);
  - the unique letters "F", "L", "O", "W", "A", "N" (6);

  hence the Jaccard index is $\frac{2}{6} \approx 33\%$.
- "STRING" and "GNIRTS" (reverse of "STRING") have all letters in common, hence the Jaccard index is 100% (perfect match).
- In the last case we see a shortcoming of Jaccard index, while for Jaro-Winkler distance 61% and for Levenshtein distance we will have a value of 0%.

Tension curve

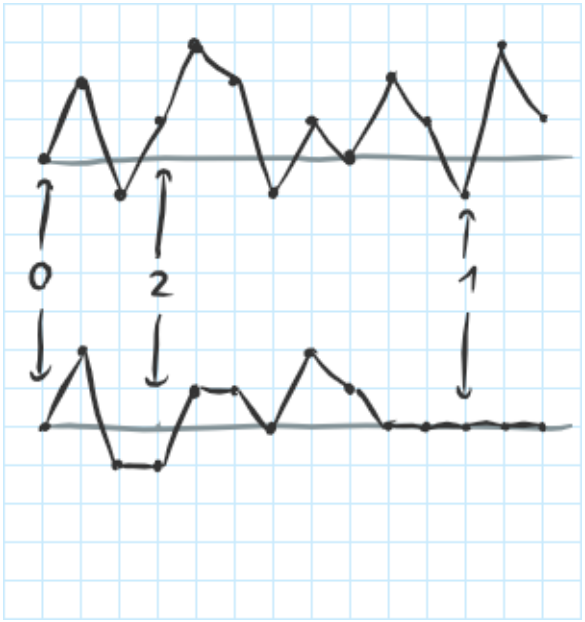
How to measure tension

- The tension curve could be extracted from character emotions, if the IS system does explicitly represent them.
- In this IS system
  1. OCC-based emotion model, which directly influences characters decision making.
  2. Every 250 ms make a snapshot of all characters' emotions, take the sum of these emotions (positive emotions $-$, negative emotions $+$), then you get the tension value at the moment $\rightarrow$ tension curve.

Tension curve
Tension distance

The distance between stories using *tension curves* is simply the distance between tension values (zero if unavailable) in this way

$$\text{distance}(x, y) = \sum_{i=1,\ldots,n} |x_i - y_i|.$$



Evaluation

Main feature: story ending

- The question is "Do stories grouped in one cluster end the same when using a particular input distance metric?".

Evaluation

Main feature: story ending

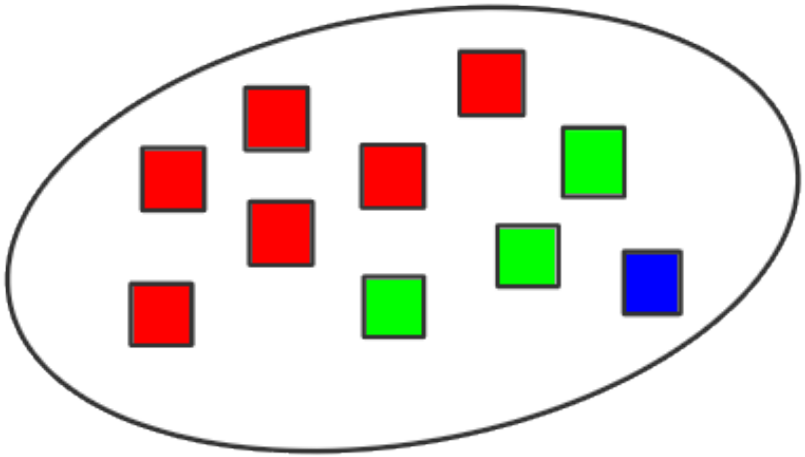
- The question is "Do stories grouped in one cluster end the same when using a particular input distance metric?".
- Used for both SD1 and SD2.
- They thought this was a suitable evaluation for a good clustering, but it is very specific to their IS systems, hence it cannot be generalized.

Evaluation

Main feature: story ending

This is represented by *precision* of the metric, i.e.

- identify the most frequent ending for each cluster;
- compute the cluster precision as the fraction of stories with this ending inside the cluster;
- the resulting precision is the weighted average of precisions of all resulting clusters.



In this case the precision is $\frac{6}{10} = 0.6$.

Table of Contents

Results
Details of the analysis

- For SD1
  - k-means with $k = 3, 4, 5$ run 8 times, results averaged.
  - The dataset is 1135 play sessions generated automatically with input being simulated by a random algorithm.

Results
Details of the analysis

- For SD1
  - k-means with $k = 3, 4, 5$ run 8 times, results averaged.
  - The dataset is 1135 play sessions generated automatically with input being simulated by a random algorithm.
- For SD2
  - k-means with $k = 4, 5, 6$ run 8 times, results averaged.
  - The dataset is 41 human play sessions, since SD2 exhibits more complex domain with more game endings.
- As a baseline: random cluster assignment, repeated 8 times, results averaged.

Level one
Results for SD1

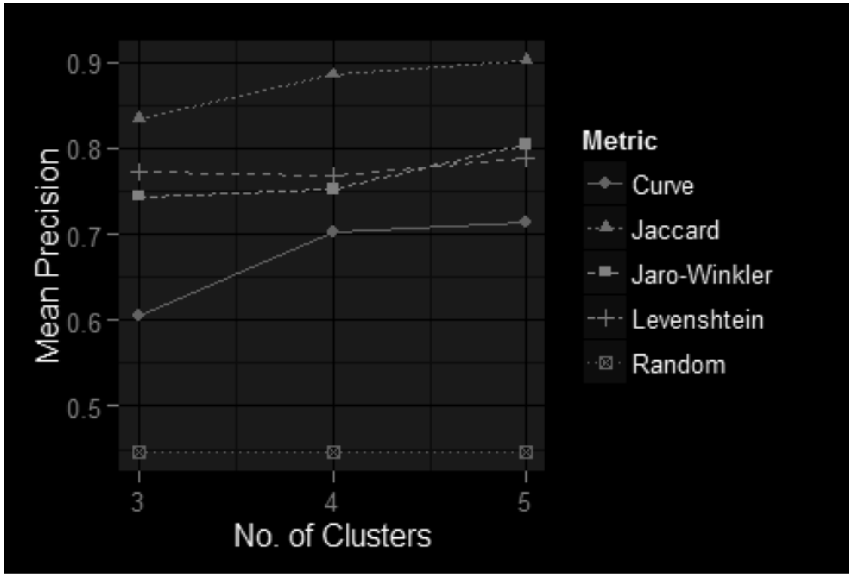
- 1135 random algorithm play sessions,
  - 608 stories ended with characters getting angry and parting;
  - 479 stories ended with characters being too positive and not reaching the cinema;
  - 16 stories ended with the characters getting to the cinema ($\rightarrow$ multi-sampling by a factor of 15);
  - 32 stories ending were undefined due to technical issues.

Level one
Results for SD1

- 1135 random algorithm play sessions,
  - 608 stories ended with characters getting angry and parting;
  - 479 stories ended with characters being too positive and not reaching the cinema;
  - 16 stories ended with the characters getting to the cinema ($\rightarrow$ multi-sampling by a factor of 15);
  - 32 stories ending were undefined due to technical issues.
- Example of two SD1 stories with breakup ending
  - "AACPDMADPDDZG00123"
  - "AABDBHHMNIOMNJ6LCDMHMCMHNIOTLDTHMNMPHPHPHPHPHPHPMHNbDTTMZG00123"

Level one
Plot of results

- The results favors the Jaccard distance metric. This was due to action sequence being very similar at the beginning and the end, since there were pre-scripted sequences after a certain event happened.
- Better to use human play sessions.



Level two
Results for SD2

- 41 human play sessions,
  - 14 stories endend with Thomas staying with Barbara;
  - 21 stories ended with Thomas staying with Nataly;
  - 4 stories ended with both girl breaking up with Thomas ($\rightarrow$ multi-sampling by a factor of two);
  - 2 stories ended with Thomas staying in relationship with both girls ($\rightarrow$ multi-sampling by a factor of two).

Level two
Results for SD2

- 41 human play sessions,
  - 14 stories endend with Thomas staying with Barbara;
  - 21 stories ended with Thomas staying with Nataly;
  - 4 stories ended with both girl breaking up with Thomas ($\rightarrow$ multi-sampling by a factor of two);
  - 2 stories ended with Thomas staying in relationship with both girls ($\rightarrow$ multi-sampling by a factor of two).
- Example of a SD2 story with "stay with Nataly" ending
  - "DBDEABACF4APDQDnAnDBABDEAqD1AkDkAkDkAkDIAIDIAIDIDQDhAMAPD1A9DQDhAqA2AgAPDQDhAgFQDQDhAgFBFQDQAqF4DEAPDL"
- In the encoding of action string, we have D for Thomas, A for Barbara and F for Nataly.

Level two
Plot of results

- Use of human play sessions is very helpful, but the sample size is indeed way too much undersized.
- The tension curve scales better than action sequences distances.

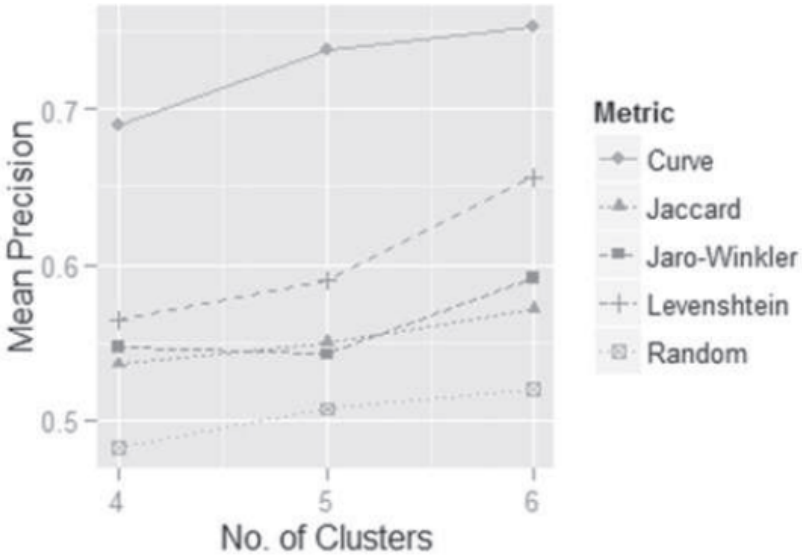


Table of Contents

Our methodology
Interactive storytelling system requirements

- The interactive story must be *story-based*, i.e. have to be represented entirely by a *story graph*, composed of *units* and connections between them.

Our methodology

Interactive storytelling system requirements

- The interactive story must be *story-based*, i.e. have to be represented entirely by a *story graph*, composed of *units* and connections between them.
- There must be a function that associates each unit (which may be annotated with emotions or texts, for instance) to a tension value, that is a numerical value representing the amount of tension in that specific unit. We will call it *tension evaluation function*.

Our methodology
Outline

1. Using a suitable explorer, extract all linear stories (or a significative sample of them, if not possible) from the story graph.

Our methodology
Outline

1. Using a suitable explorer, extract all linear stories (or a significative sample of them, if not possible) from the story graph.
2. Using the tension evaluation function, each linear story has a corresponding tension curve.

Our methodology
Outline

1. Using a suitable explorer, extract all linear stories (or a significative sample of them, if not possible) from the story graph.
2. Using the tension evaluation function, each linear story has a corresponding tension curve.
3. Using the tension curve metric, run a k-means clustering algorithm on the linear stories.

Our methodology
Outline

1. Using a suitable explorer, extract all linear stories (or a significative sample of them, if not possible) from the story graph.
2. Using the tension evaluation function, each linear story has a corresponding tension curve.
3. Using the tension curve metric, run a k-means clustering algorithm on the linear stories.
4. Design a user survey, so that every linear story has a human score associated that express roughly the quality of the story.

Our methodology
Outline

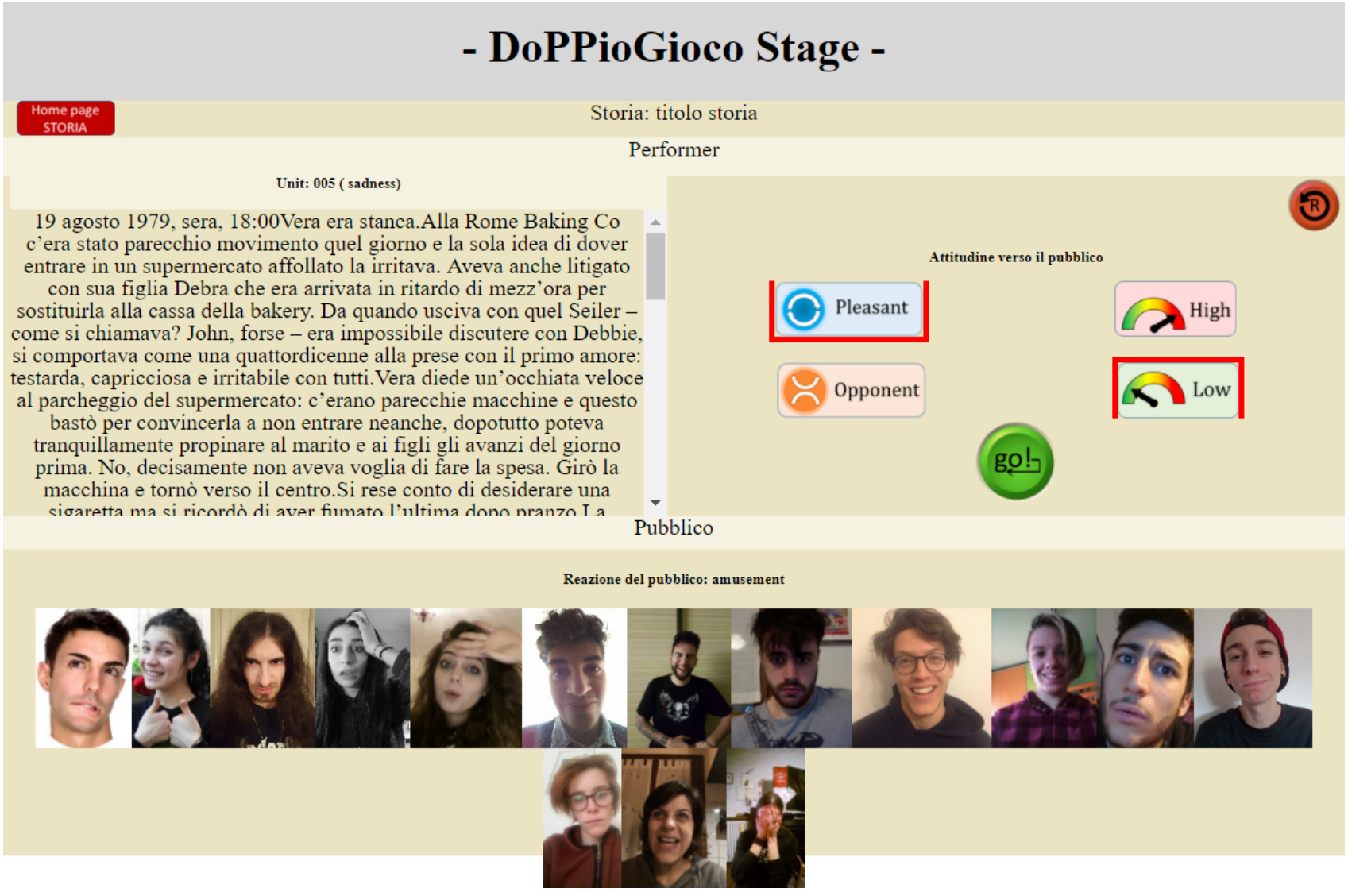
1. Using a suitable explorer, extract all linear stories (or a significative sample of them, if not possible) from the story graph.
2. Using the tension evaluation function, each linear story has a corresponding tension curve.
3. Using the tension curve metric, run a k-means clustering algorithm on the linear stories.
4. Design a user survey, so that every linear story has a human score associated that express roughly the quality of the story.
5. Use the quality scores to evaluate the clustering.

Our methodology
Outline

1. Using a suitable explorer, extract all linear stories (or a significative sample of them, if not possible) from the story graph.
2. Using the tension evaluation function, each linear story has a corresponding tension curve.
3. Using the tension curve metric, run a k-means clustering algorithm on the linear stories.
4. Design a user survey, so that every linear story has a human score associated that express roughly the quality of the story.
5. Use the quality scores to evaluate the clustering.
6. The story designer can inspect the clusters to have insights into the IS system.

DoppioGioco

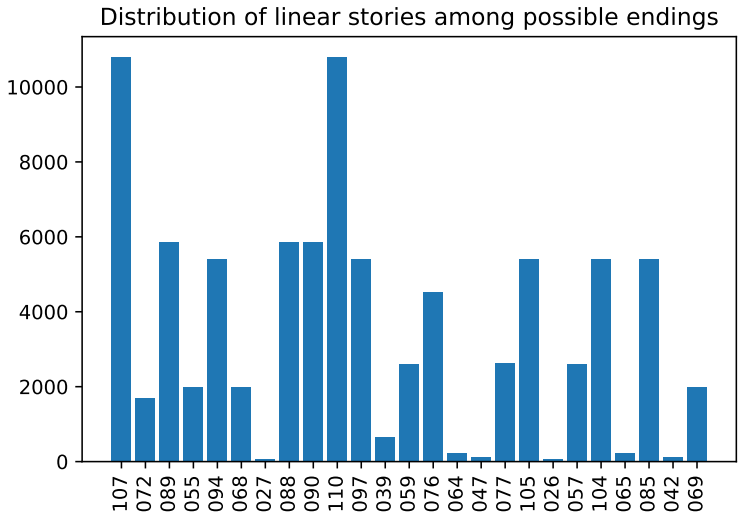
Our IS of choice



DoppioGioco

Some statistics

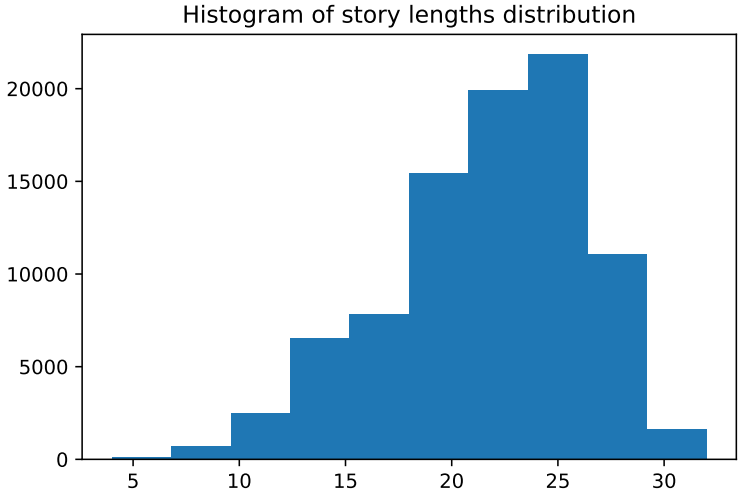
Numer of units	104
Possible linear stories	87 686
Story beninnings	1
Story endings	25
Story length average	21.60
Story length mode	24
Story length max	32
Story length min	4



DoppioGioco

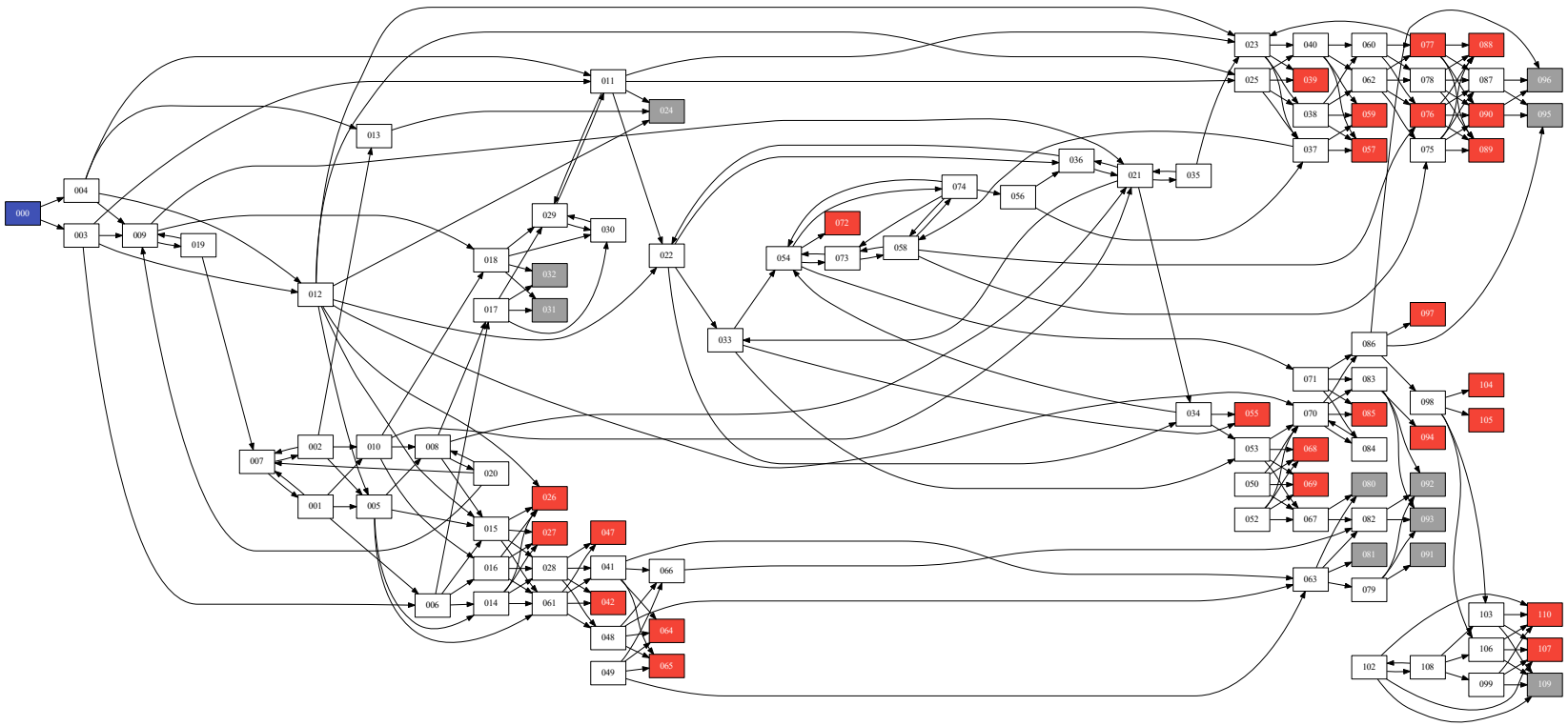
Some statistics

Numer of units	104
Possible linear stories	87 686
Story beninnings	1
Story endings	25
Story length average	21.60
Story length mode	24
Story length max	32
Story length min	4



DoppioGioco

Story graph representation



Tension evaluation function

GEMEP emotion model

- The emotions in DoppioGioco are encoded using the GEMEP emotion model.
- Emotions are divided into four main categories, whether they have positive or negative valence and whether they ignites high or low arousal.
  1. *positive valence, high arousal*: joy (elation), amusement, pride.

Tension evaluation function

GEMEP emotion model

- The emotions in DoppioGioco are encoded using the GEMEP emotion model.
- Emotions are divided into four main categories, whether they have positive or negative valence and whether they ignites high or low arousal.
  1. *positive valence, high arousal*: joy (elation), amusement, pride.
  2. *positive valence, low arousal*: pleasure, relief, interest.

Tension evaluation function

GEMEP emotion model

- The emotions in DoppioGioco are encoded using the GEMEP emotion model.
- Emotions are divided into four main categories, whether they have positive or negative valence and whether they ignites high or low arousal.
  1. *positive valence, high arousal*: joy (elation), amusement, pride.
  2. *positive valence, low arousal*: pleasure, relief, interest.
  3. *negative valence, low arousal*: cold anger (irritation), anxiety (worry), sadness (depression).

Tension evaluation function

GEMEP emotion model

- The emotions in DoppioGioco are encoded using the GEMEP emotion model.
- Emotions are divided into four main categories, whether they have positive or negative valence and whether they ignites high or low arousal.
  1. *positive valence, high arousal*: joy (elation), amusement, pride.
  2. *positive valence, low arousal*: pleasure, relief, interest.
  3. *negative valence, low arousal*: cold anger (irritation), anxiety (worry), sadness (depression).
  4. *negative valence, high arousal*: hot anger (rage), panic fear, despair.

Tension evaluation function

How we can evaluate the tension value

- One naive yet appropriate way to measure tension is that of assigning a higher value to high arousal emotions (e.g. 2) and a lower value to low arousal emotions (e.g. 1). Then, the positive or negative valence determines the sign of the value. A zero value is assigned to non-annotated units.
  1. *positive valence, high arousal*: $+2$.
  2. *positive valence, low arousal*: $+1$.
  3. *negative valence, low arousal*: -1.
  4. *negative valence, high arousal*: -2.
- Other more complex tension evaluation functions can be tested if results with these tension values are not satisfying.

Tension evaluation function

Example of a tension curve

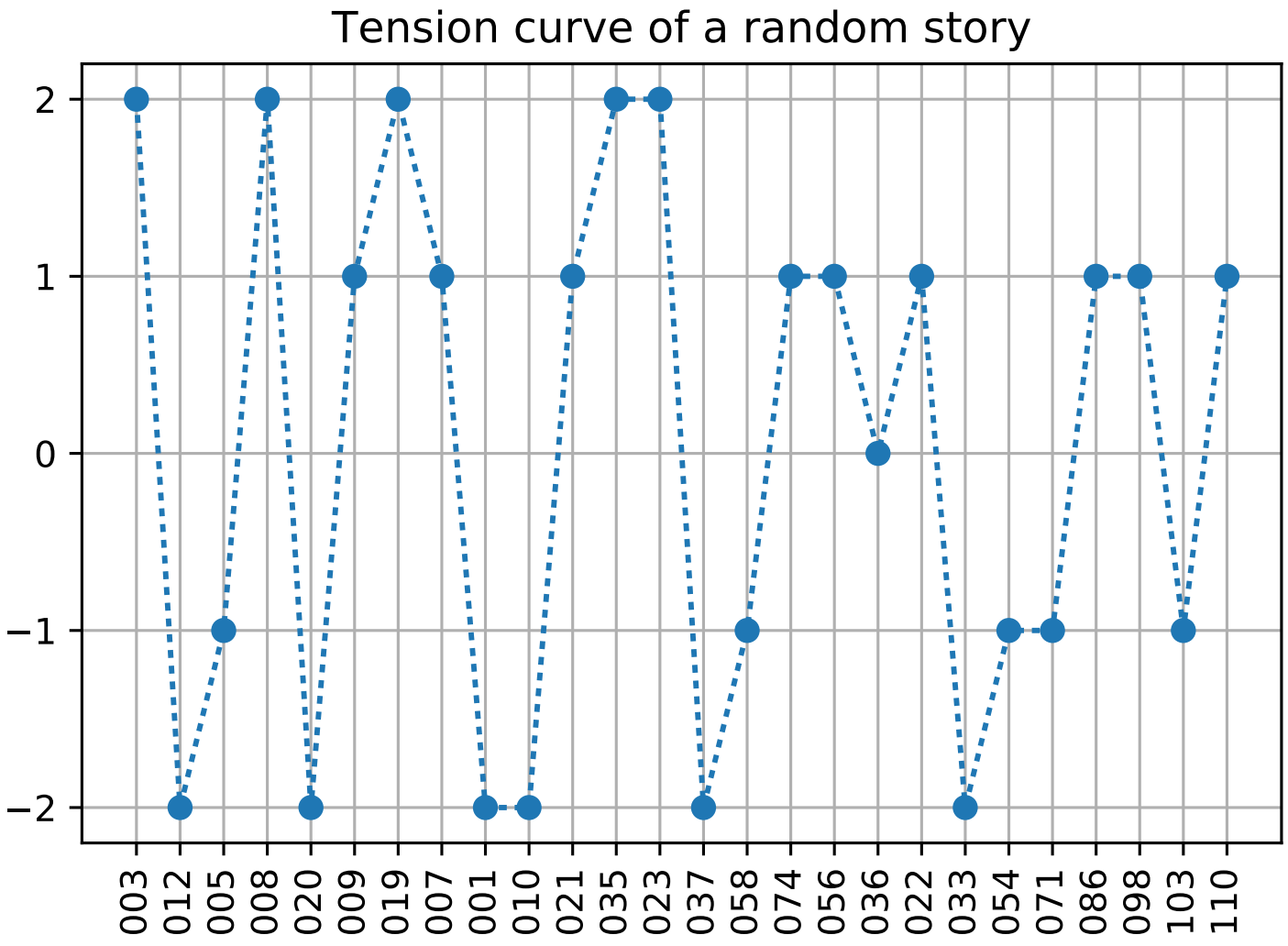


Table of Contents

Story designer perspective

What a story designer may discover about its IS

- The output of the algorithm will be a set of clusters (i.e. groups), each with a peculiar tension curve shape and an average quality score.

Story designer perspective

What a story designer may discover about its IS

- The output of the algorithm will be a set of clusters (i.e. groups), each with a peculiar tension curve shape and an average quality score.
- The story designer may inspect good quality score clusters and try to understand what makes them better with respect to the others.

Story designer perspective

What a story designer may discover about its IS

- The output of the algorithm will be a set of clusters (i.e. groups), each with a peculiar tension curve shape and an average quality score.
- The story designer may inspect good quality score clusters and try to understand what makes them better with respect to the others.
- The story designer may inspect bad quality score clusters and decide whether to leave them as they are, review them or remove them from the IS system.

Interactive storytelling systems perspective

What interactive story designers may discover about ISs

- For linear stories there are studies that show a correlation between popular stories and Freytag pyramid.

Interactive storytelling systems perspective

What interactive story designers may discover about ISs

- For linear stories there are studies that show a correlation between popular stories and Freytag pyramid.
- May there be some tension curves that does as well for interactive stories? Or maybe Freytag pyramid is still a good reference?

Interactive storytelling systems perspective

What interactive story designers may discover about ISs

- For linear stories there are studies that show a correlation between popular stories and Freytag pyramid.
- May there be some tension curves that does as well for interactive stories? Or maybe Freytag pyramid is still a good reference?
- This is not much likely, but if a cluster shows a specific tension curve shape with a incredibly high quality score, one may infer that such shape is a good indicator of a good story. From this, it would be possible to analyze other IS systems and check if the same behaviour happens.

Risks and possibilities

Possible problems to overcome

- If clusters are not very different, e.g. they have almost all the same score, they give no information at all.

Risks and possibilities

Possible problems to overcome

- If clusters are not very different, e.g. they have almost all the same score, they give no information at all.
- If tension curves are too different or not at all, the clustering will be again a bad one.

Risks and possibilities

Possible problems to overcome

- If clusters are not very different, e.g. they have almost all the same score, they give no information at all.
- If tension curves are too different or not at all, the clustering will be again a bad one.
- To avoid these problems, we must design carefully the user survery and choose cautiosly the tension evaluation function.

Risks and possibilities

Possible problems to overcome

- If clusters are not very different, e.g. they have almost all the same score, they give no information at all.
- If tension curves are too different or not at all, the clustering will be again a bad one.
- To avoid these problems, we must design carefully the user survery and choose cautiosly the tension evaluation function.
- DoppioGioco is not at all a professionally designed interactive storytelling system. It has its flaws, that can be hopefully detected by the algorithm.

Conclusion

- This algorithm will be a step forward into the almost unknown field of interactive storytelling.

Conclusion

- This algorithm will be a step forward into the almost unknown field of interactive storytelling.
- Since IS systems can be very complex, a computer assisted approach would be helpful to story designers.

Conclusion

- This algorithm will be a step forward into the almost unknown field of interactive storytelling.
- Since IS systems can be very complex, a computer assisted approach would be helpful to story designers.
- We hope to build a tool to help making amazing interactive stories easier!