GRAPH-BASED FEATURES FOR ONLINE AUTOMATIC ABUSE DETECTION

5th SLSP Conference
Statistical Language and Speech Processing
Le Mans, France, October 23-25 2017

Etienne Papegnies[1,2], Richard Dufour[1], Vincent Labatut[1] & Georges Linarès[1]

firstname.lastname@univ-avignon.fr
1 : LIA EA 4128 – Université d'Avignon et des Pays de Vaucluse
2 : Nectar de Code, Barbentane

OVERVIEW

1. Context
2. Existing approaches
3. Method
4. Results
5. Conclusions & perspectives

ONLINE COMMUNITIES & MODERATION

- Online Communities
  - Important Medium : widely used, high socio-economical impact
  - Users are usually anonymous

ONLINE COMMUNITIES & MODERATION

- Online Communities
  - Important Medium : widely used, high socio-economical impact
  - Users are usually anonymous
- Abusive Behavior
  - Violation of the rules of the community
  - Can lead to : community degradation, legal consequences

ONLINE COMMUNITIES & MODERATION

- Online Communities
  - Important Medium : widely used, high socio-economical impact
  - Users are usually anonymous
- Abusive Behavior
  - Violation of the rules of the community
  - Can lead to : community degradation, legal consequences
- Moderation
  - Detecting abusive users and applying sanctions
  - Usually done by hand : costly

AUTOMATIZED MODERATION

- Automation
  - Assistance : raise messages to moderator’s attention
  - Full moderation : detect abuse and apply sanctions

AUTOMATIZED MODERATION

- Automation
  - Assistance : raise messages to moderator's attention
  - Full moderation : detect abuse and apply sanctions
  - Problem is not trivial (ex. Google Perspective API)
    - Noise (Can be intentional)
    - Natural Language
    - Context

AUTOMATIZED MODERATION

- Automation
  - Assistance : raise messages to moderator's attention
  - Full moderation : detect abuse and apply sanctions
  - Problem is not trivial (ex. Google Perspective API)
    - Noise (Can be intentional)
    - Natural Language
    - Context
- In this work :
  - Detection of abusive messages : binary classification task
  - We use features extracted from a graph representation of the conversation surrounding a message
  - Applied to data from the MMORPG SpaceOrigin

ABUSE DETECTION

- Content-Based Approaches [Spe97, CZZX12, DRL11, CS15]
  - Badwords dictionaries
  - Static rules
  - Word n-gram approaches
  - Bag of Words models (*tf–idf*)

ABUSE DETECTION

- Content-Based Approaches [Spe97, CZZX12, DRL11, CS15]
  - Badwords dictionaries
  - Static rules
  - Word n-gram approaches
  - Bag of Words models (*tf–idf*)
- Context-Based approaches
  [YXH+09, CDNML15, BS15, GDFMGM16]
  - Content of neighboring messages
  - User models (language, behavior)
  - Interactions outside of discussions

ABUSE DETECTION

- Content-Based Approaches [Spe97, CZZX12, DRL11, CS15]
  - Badwords dictionaries
  - Static rules
  - Word n-gram approaches
  - Bag of Words models (*tf–idf*)
- Context-Based approaches
  [YXH+09, CDNML15, BS15, GDFMGM16]
  - Content of neighboring messages
  - User models (language, behavior)
  - Interactions outside of discussions
- CICLing'17 [PLDL17]
  - Specific Preprocessing (ex. reversal of hex or binary coding)
  - Morphological Features : character counts, compression rate
  - Language Features : *tf–idf*, word / named entity counts, sentiment score...
  - Behavioral Features : response strength, user language models...

EXTRACTION OF CONVERSATIONAL NETWORKS

- Data : raw chat logs

EXTRACTION OF CONVERSATIONAL NETWORKS

- Data : raw chat logs
- Objectives
  - Visualize interactions
  - Identify User Classes, Roles

EXTRACTION OF CONVERSATIONAL NETWORKS

- Data : raw chat logs
- Objectives
  - Visualize interactions
  - Identify User Classes, Roles
- Problem : who's the intended recipient for a message ?
  - Direct referencing (some flexibility) [Mut04, TMZ14, SR14]
  - Links to every possible recipients [TMZ14]
  - Predefined rules to identify recipients [Mut04, TMZ14]
  - Proximity and temporal density of messages [Mut04]
  - Thread detection by content analysis [TMR10]

METHOD

1. Conversational Network Extraction

METHOD

1. Conversational Network Extraction
    - Weighted non-directional graph
    - Build around a *target message*
    - Spawns a pre-defined *context period*
    - Vertices : active users within the context period
    - Links : message-based interactions between users
    - Weights : intensity of the interaction

METHOD

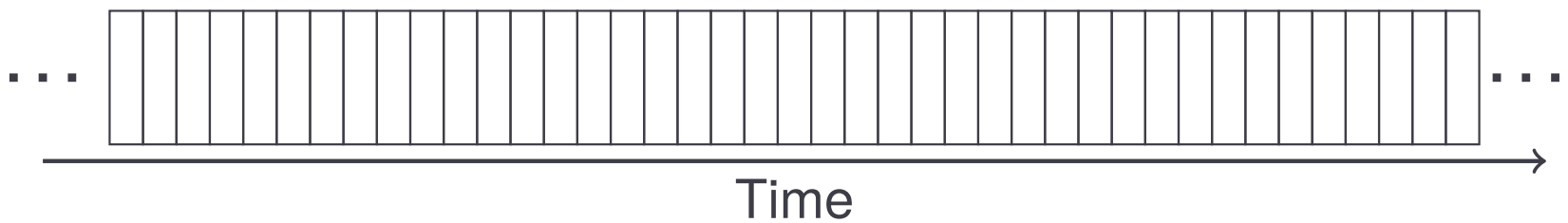
1. Conversational Network Extraction
    - Weighted non-directional graph
    - Build around a *target message*
    - Spawns a pre-defined *context period*
    - Vertices : active users within the context period
    - Links : message-based interactions between users
    - Weights : intensity of the interaction
2. Compute topological measures

METHOD

1. Conversational Network Extraction
   - Weighted non-directional graph
   - Build around a *target message*
   - Spawns a pre-defined *context period*
   - Vertices : active users within the context period
   - Links : message-based interactions between users
   - Weights : intensity of the interaction
2. Compute topological measures
3. SVM training

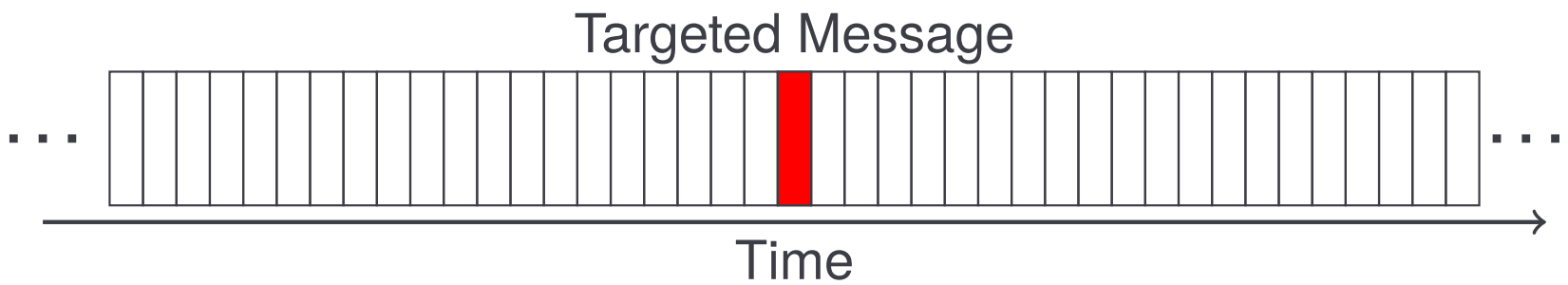
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



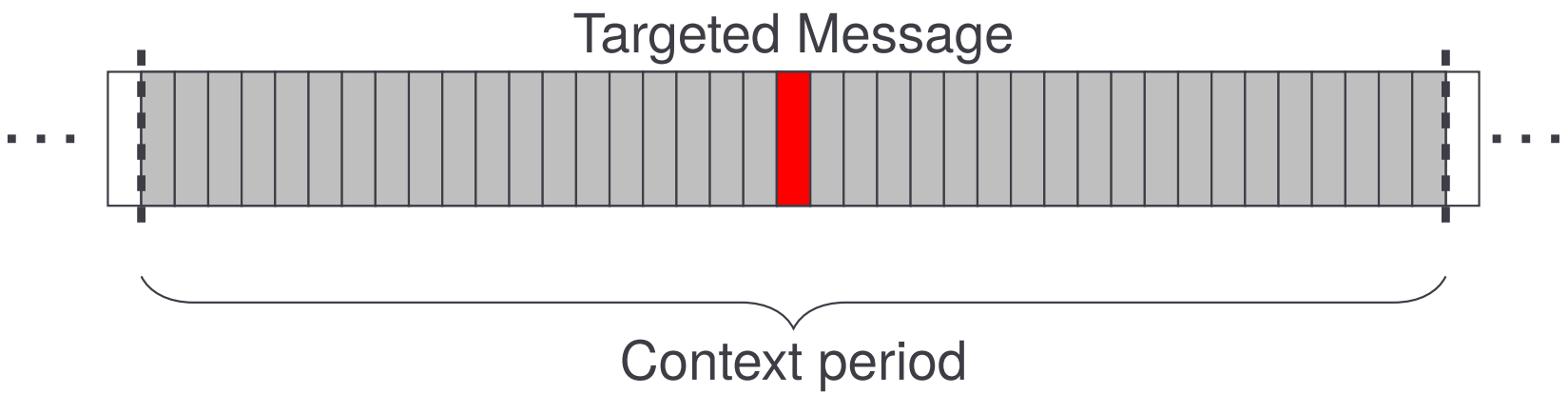
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



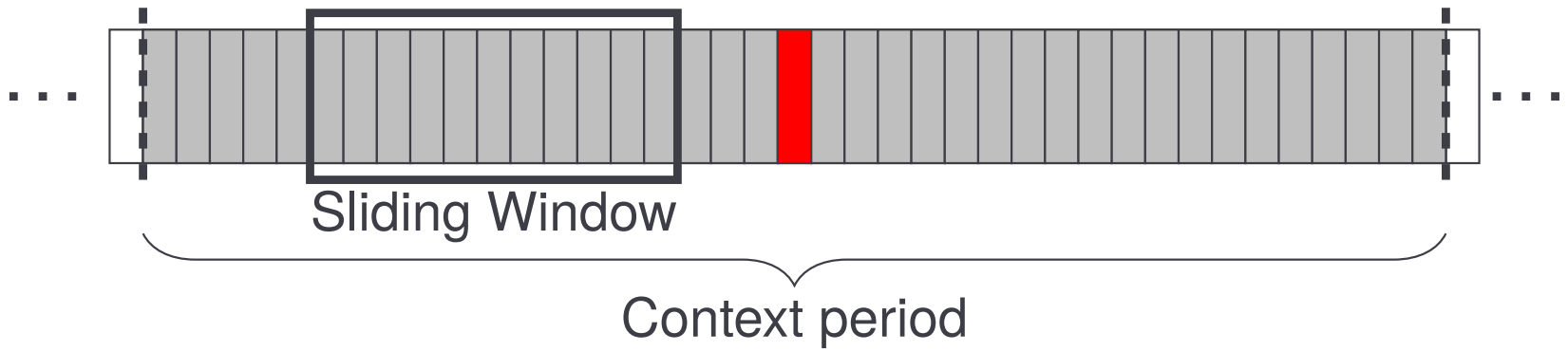
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



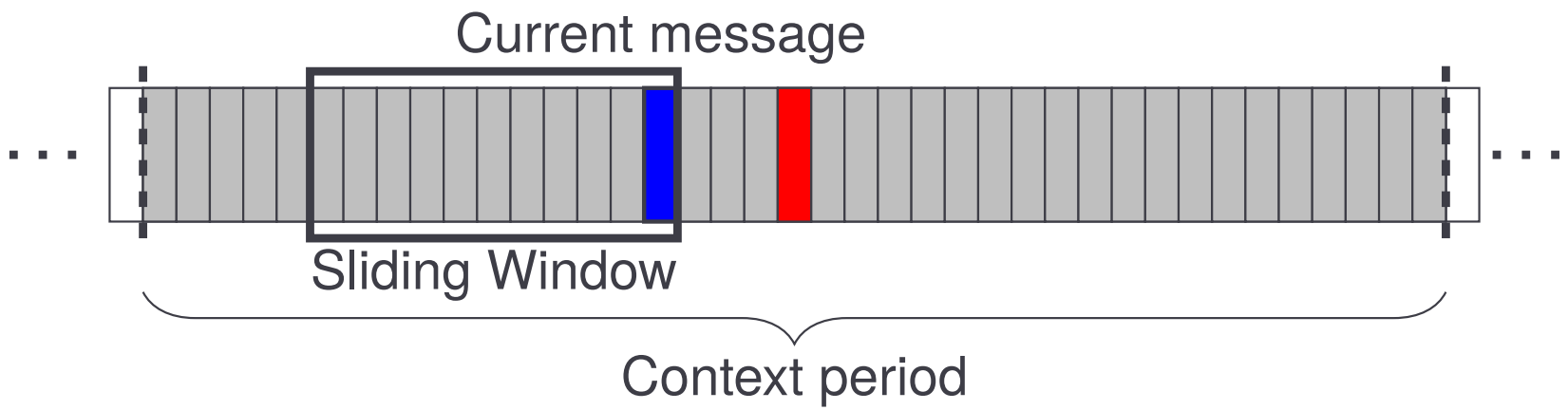
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



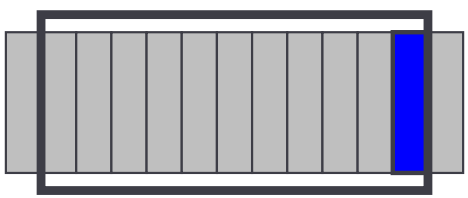
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



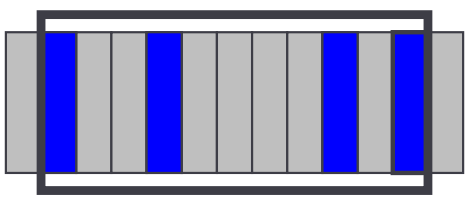
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
    - Hyp. #1 : current message targeted towards other participants
    - Hyp. #2 : message addressed to last seen users first
    - Hyp. #3 : directly referenced users even more targeted



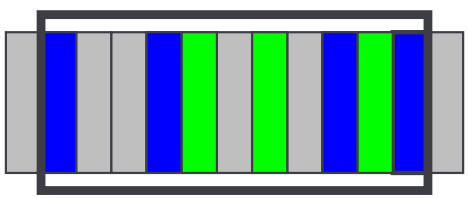
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
    - Hyp. #1 : current message targeted towards other participants
    - Hyp. #2 : message addressed to last seen users first
    - Hyp. #3 : directly referenced users even more targeted



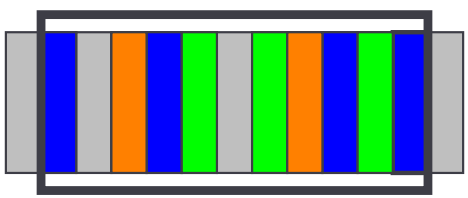
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
    - Hyp. #1 : current message targeted towards other participants
    - Hyp. #2 : message addressed to last seen users first
    - Hyp. #3 : directly referenced users even more targeted



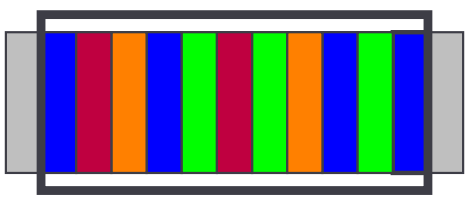
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



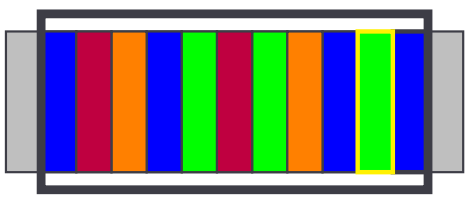
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



CONVERSATIONAL NETWORK EXTRACTION

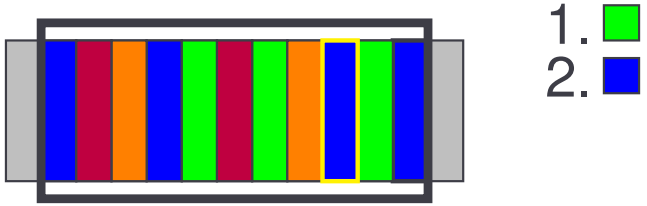
1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



1. ■

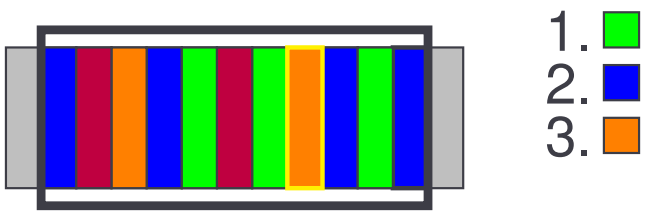
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



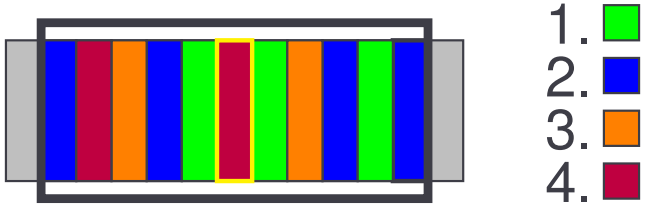
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



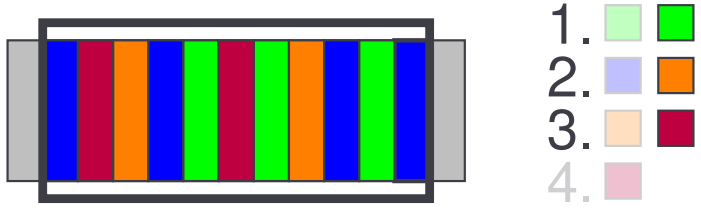
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



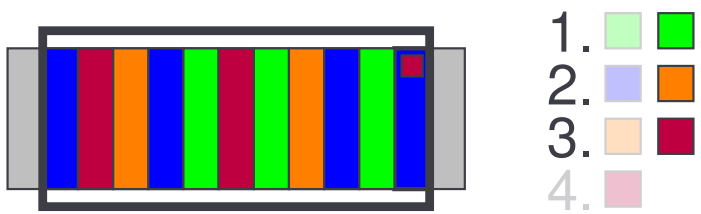
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



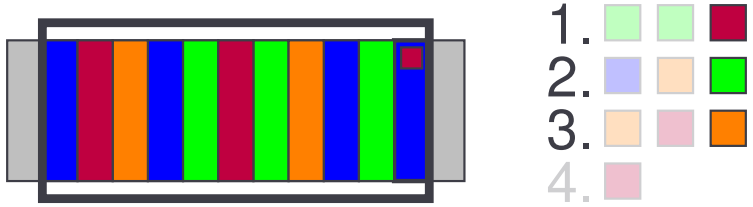
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



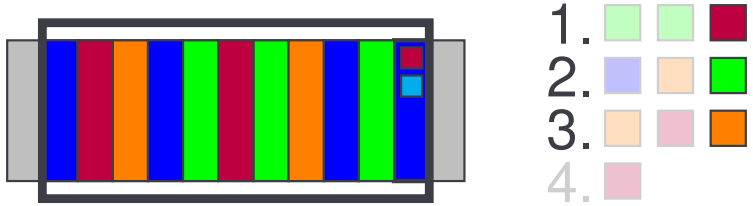
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



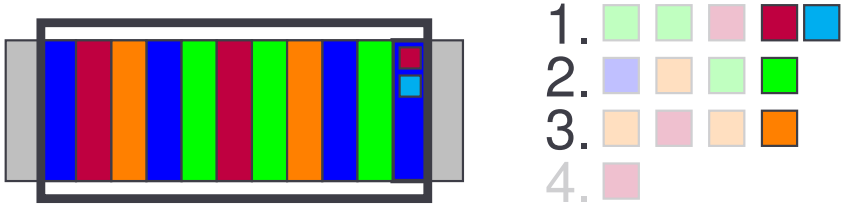
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



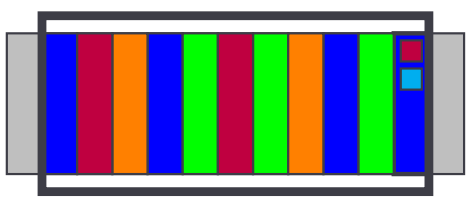
CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



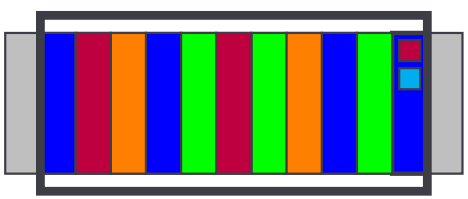
CONVERSATIONAL NETWORK EXTRACTION

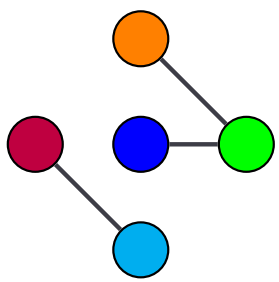
1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted



CONVERSATIONAL NETWORK EXTRACTION

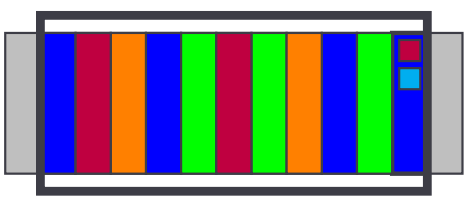
1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted
4. Update the graph

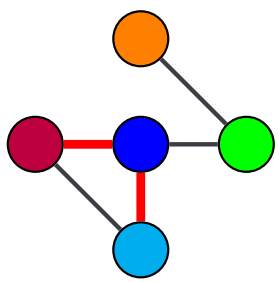




CONVERSATIONAL NETWORK EXTRACTION

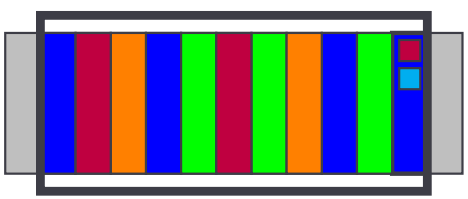
1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted
4. Update the graph

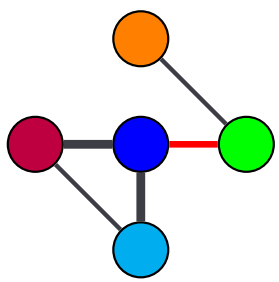




CONVERSATIONAL NETWORK EXTRACTION

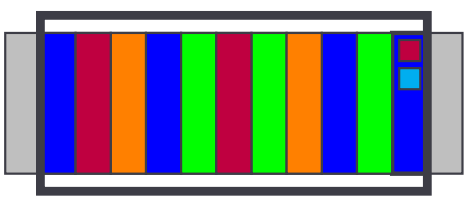
1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted
4. Update the graph

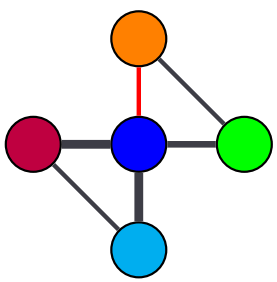




CONVERSATIONAL NETWORK EXTRACTION

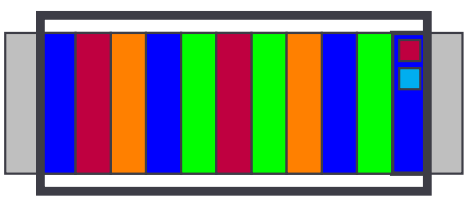
1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted
4. Update the graph

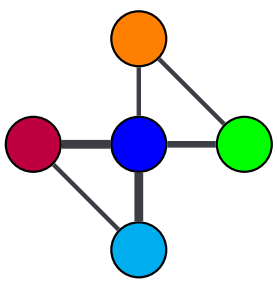




CONVERSATIONAL NETWORK EXTRACTION

1. Define context period, centered on target message
2. Slide window over conversation relative to a *current message*
3. Compute weights of links
   - Hyp. #1 : current message targeted towards other participants
   - Hyp. #2 : message addressed to last seen users first
   - Hyp. #3 : directly referenced users even more targeted
4. Update the graph





TOPOLOGICAL MEASURES

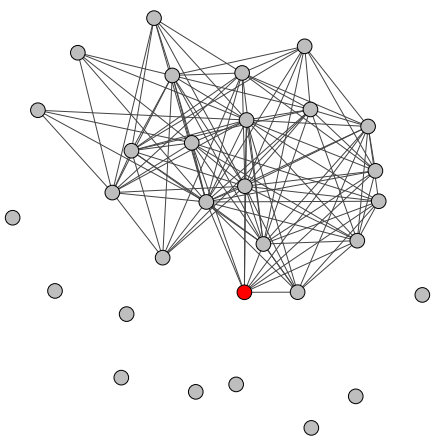
- Local measures
  - Degree, Eigenvector, PageRank, Hub & Authority
  - Betweeness, Closeness, Eccentricity, Coreness

TOPOLOGICAL MEASURES

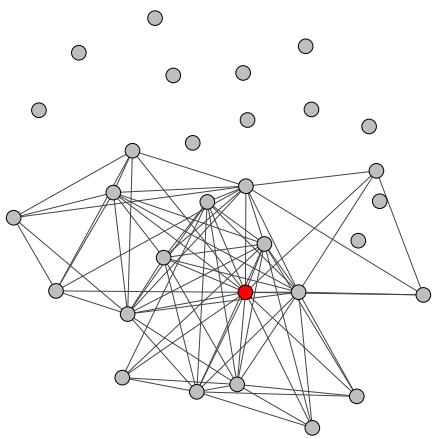
- Local measures
  - Degree, Eigenvector, PageRank, Hub & Authority
  - Betweeness, Closeness, Eccentricity, Coreness
- Global measures
  - Vertices/Edges numbers, density
  - Diameter, average distance
  - Number of Cliques
  - Degree Assortativity
  - Averages of each local measure over the whole network

TOPOLOGICAL MEASURES

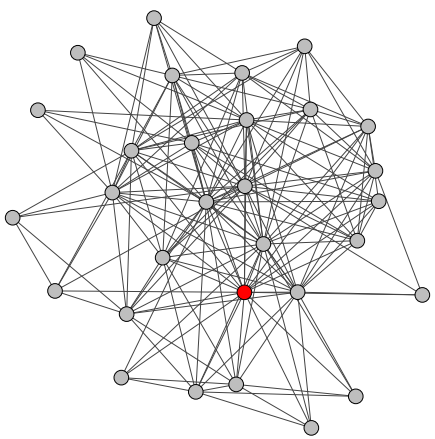
- Local measures
  - Degree, Eigenvector, PageRank, Hub & Authority
  - Betweeness, Closeness, Eccentricity, Coreness
- Global measures
  - Vertices/Edges numbers, density
  - Diameter, average distance
  - Number of Cliques
  - Degree Assortativity
  - Averages of each local measure over the whole network
- Three networks for each targeted message



Before



After



Full

DATASET & EXPERIMENTAL PROTOCOL

- Dataset
  - Chat logs from the MMORPG SpaceOrigin
  - 4 029 343 instant messages
    - 779 messages flagged and later confirmed as abusive
    - Sample of 2 000 messages assumed non-abusive
    - All messages taken from different conversations

DATASET & EXPERIMENTAL PROTOCOL

- Dataset
  - Chat logs from the MMORPG SpaceOrigin
  - 4 029 343 instant messages
    - 779 messages flagged and later confirmed as abusive
    - Sample of 2 000 messages assumed non-abusive
    - All messages taken from different conversations

- Classification
  - SVM (Sklearn C-Support Vector Classification)
  - Cross validation with 70–30% split
  - Feature importance estimated using ExtraTreesClassifier (Sklearn)

CLASSIFICATION RESULTS

Scores relative to the *Abuse* class

Classifier	**Precision**	**Recall**	***F*-measure**
Random Baseline	0,28	0,50	0,36
Text-Based content/context features [PLDL17]	0,70	0,74	0,72
Graph Features	0,77	0,77	0,77

CLASSIFICATION RESULTS

Scores relative to the *Abuse* class

Classifier	**Precision**	**Recall**	***F*-measure**
Random Baseline	0,28	0,50	0,36
Text-Based content/context features [PLDL17]	0,70	0,74	0,72
Graph Features	0,77	0,77	0,77

- Better performances even while completely ignoring content
  - Possible reasons : better usage of post-abuse information

CLASSIFICATION RESULTS

Scores relative to the *Abuse* class

Classifier	**Precision**	**Recall**	***F*-measure**
Random Baseline	0,28	0,50	0,36
Text-Based content/context features [PLDL17]	0,70	0,74	0,72
Graph Features	0,77	0,77	0,77

- Better performances even while completely ignoring content
  - Possible reasons : better usage of post-abuse information
- Classifier can be used to assist in moderation

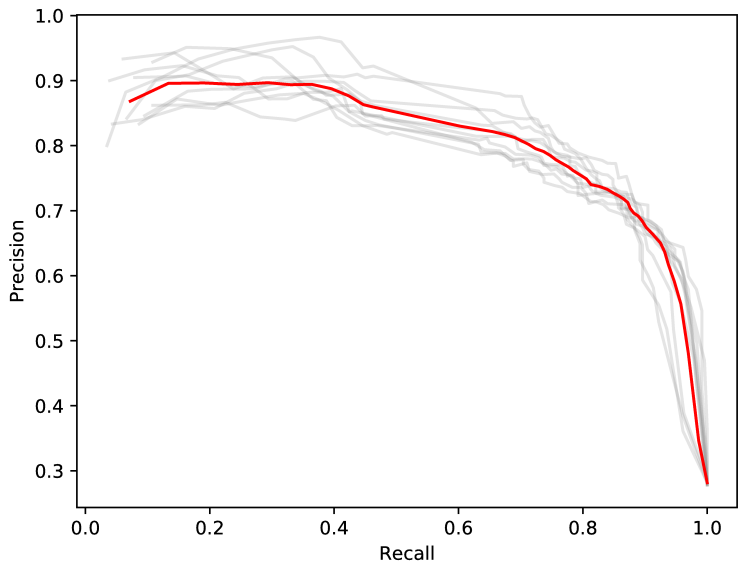


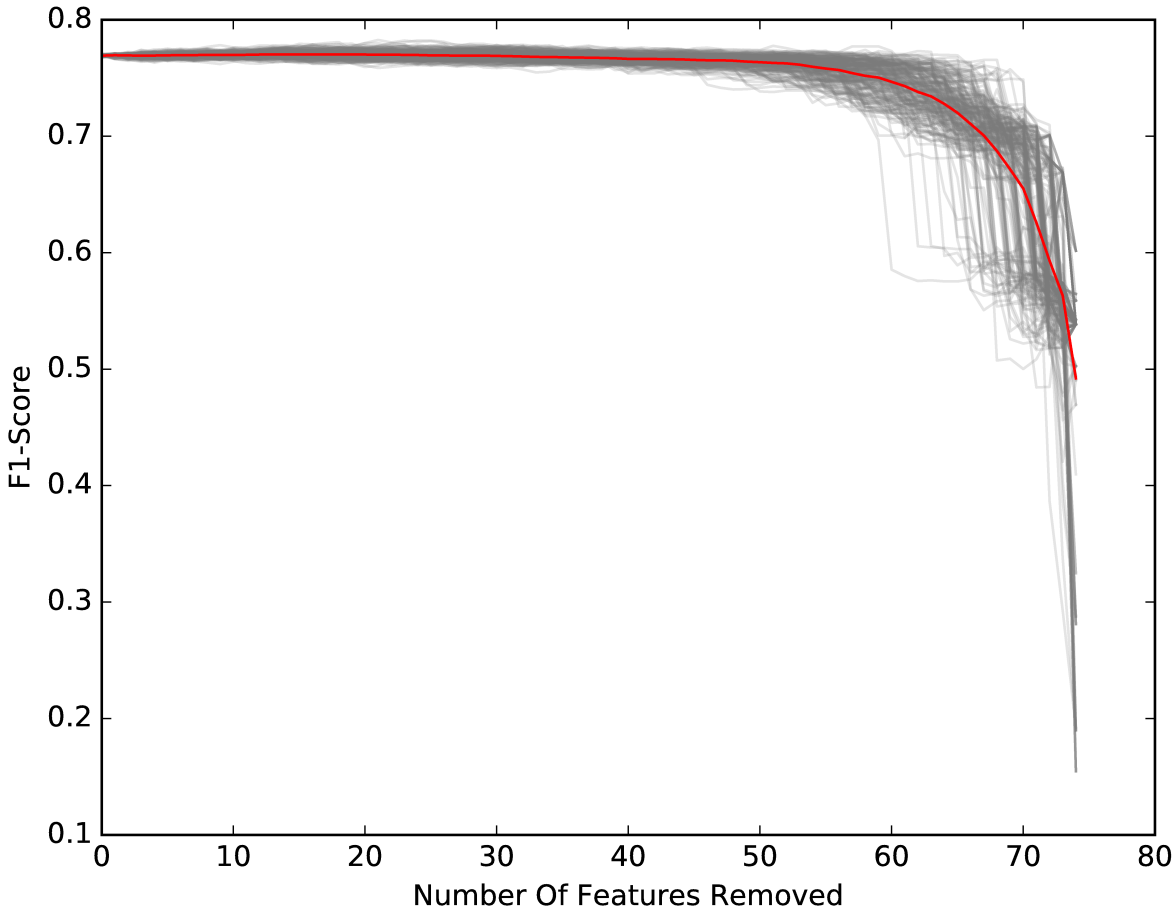
FIGURE – Precision-Recall curves for the 10 classifiers

FEATURE SELECTION

- Method : successive ablation of features

FEATURE SELECTION

- Method : successive ablation of features



FEATURE SELECTION

- Method : successive ablation of features

Most discriminating features in the Graph-based approach

Graph	**Feature**	***F*-measure *before* ablation**
Full	Average Betweenness	0,76
Before	Average Coreness	0,75
After	Edge Number	0,75
After	Density	0,73
Full	Hub Score	0,73
After	Degree Centrality	0,68
Before	Vertice Number	0,67
Full	Average Eccentricity	0,58
Before	Average Eigenvector	0,57
Full	Eccentricity	0,35

FEATURE SELECTION

- Method : successive ablation of features

  Most discriminating features in the Graph-based approach

Graph	Feature	*F*-measure ***before*** ablation
Full	Average Betweenness	0,76
Before	Average Coreness	0,75
After	Edge Number	0,75
After	Density	0,73
Full	Hub Score	0,73
After	Degree Centrality	0,68
Before	Vertice Number	0,67
Full	Average Eccentricity	0,58
Before	Average Eigenvector	0,57
Full	Eccentricity	0,35

- Observations
  - Important measures characterize the graph in different ways.
  - Some measures belong to high correlation groups and can be swapped
  - Considering the two sides (before / after) yields better discrimination

CONCLUSIONS & PERSPECTIVES

- Main results
  - Simple approach
  - Robust with regard to text preprocessing issues
  - Results are better than with our text-based approach
  - Performance is good enough to provide support, not full automation
  - Limits : computational cost, no real-time application

CONCLUSIONS & PERSPECTIVES

- Main results
  - Simple approach
  - Robust with regard to text preprocessing issues
  - Results are better than with our text-based approach
  - Performance is good enough to provide support, not full automation
  - Limits : computational cost, no real-time application
- Perspectives
  - Tweak parameters used for network extraction
  - Use different graph measures
  - Combine the approach with the text-based one
  - Explore combinations and individual contributions of the three graphs
  - Dynamic network modeling
  - User model profiles

Questions

[BS15] K. Balci and A. A. Salah.
Automatic analysis and identification of verbal aggression and abusive behaviors for online social games.
Computers in Human Behavior, 53 :517–526, 2015.

[CDNML15] J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec.
Antisocial behavior in online discussion communities.
arXiv :1504.00680 [cs.SI], 2015.

[CS15] V. S. Chavan and S. S. Shylaja.
Machine learning approach for detection of cyber-aggressive comments by peers on social media network.
In *IEEE ICACCI*, pages 2354–2358, 2015.

[CZZX12] Y. Chen, Y. Zhou, S. Zhu, and H. Xu.
Detecting offensive language in social media to protect adolescent online safety.
In *PASSAT/SocialCom*, pages 71–80, 2012.

[DRL11] K. Dinakar, R. Reichart, and H. Lieberman.
Modeling the detection of textual cyberbullying.
5th International AAAI Conference on Weblogs and Social Media, pages 11–17, 2011.

[GDFMGM16] K. Garimella, G. De Francisci Morales, A. Gionis, and M. Mathioudakis.
Quantifying controversy in social media.
In *9th ACM International Conference on Web Search and Data Mining*, pages 33–42, 2016.

[Mut04] P. Mutton.
Inferring and visualizing social networks on internet relay chat.
In *8th International Conference on Information Visualisation*, pages 35–43, 2004.

[PLDL17] E. Papegnies, V. Labatut, R. Dufour, and G. Linarès.
Impact of content features for automatic online abuse detection.
In *International Conference on Computational Linguistics and Intelligent Text Processing*, 2017.

[Spe97] E. Spertus.
Smokey : Automatic recognition of hostile messages.
In *14th National Conference on Artificial Intelligence and 9th Conference on Innovative Applications of Artificial Intelligence*, pages 1058–1065, 1997.

[SR14] T. Sinha and I. Rajasingh.
Investigating substructures in goal oriented online communities : Case study of Ubuntu IRC.
In *IEEE International Advance Computing Conference*, pages 916–922, 2014.

[TMR10] S. Trausan-Matu and T. Rebedea.
A polyphonic model and system for inter-animation analysis in chat conversations with multiple participants.
In *Computational Linguistics and Intelligent Text Processing*, volume 6008 of *Lecture Notes in Computer Science*, pages 354–363. Springer, 2010.

[TMZ14] S. Tavassoli, M. Moessner, and K. A. Zweig.
Constructing social networks from semi-structured chat-log data.
In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 146–149, 2014.

[YXH+09] D. Yin, Z. Xue, L. Hong, B. D. Davison, A. Kontostathis, and L. Edwards.
Detection of harassment on Web 2.0.
In *WWW Workshop : Content Analysis in the Web 2.0*, 2009.

IMPACT OF LENGTH OF CONTEXT PERIOD