Sentiment Analysis using Word-Graphs

SUPER

Social sensors for secUrity assessments and Proactive EmeRgencies management

John Violos, Konstantinos Tserpes, Evangelos Psomakelis, Konstantinos Psychas, Theodora Varvarigou

National Technical University of Athens

WIMS' 16 6th international conference on Web-Intelligence, Mining and Semantics

15 / 06 / 2016

SUPER Project

Co-funded by the European Commission's 7th Framework Programme

SUPER is a joint effort of social media and emergency management experts towards introducing a holistic, integrated and privacy-friendly approach to the use of social media in emergencies and security incidents.

Hurricane Sandy 2012 Oct 27 – Nov 1: 20M + Tweets

Picture from: www.theworld4realz.com

Social Sensors

- Event - SubEvent Detection
- Topic Community Tracking
- Sentiment Analysis
- Behaviour Abalysis
- Rumour Spreading Identification – Credibility
- Intelligent Fusion and Reasoning

Improve-Modify

Probabilistic Method Component

Deep Learning Component

Word Graph Representation Component

N

Sentiment Analysis 1/2

What is our principal challenge?

To detect Sentiment Polarities of published textual posts in SN!

Sentiments:

+ Positive

- Negative

= Neutral

Sentiment Analysis 2/2

How?

SA Problem → Text Classification Problem

What is our Contribution?

- **Different Reprsentation Model (No BoW)**
- **Similarity Measures**

Valuable information:

Neighborhood of the Words

Sequence of words

Example:

+

The Movie is not boring
I do like it

-

I do not like the Movie
It is boring

Same Set of Words Different Sentiments

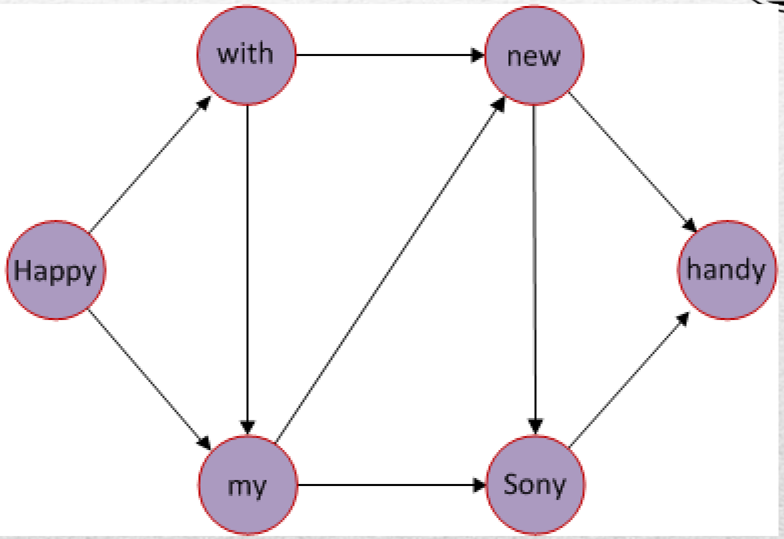
A good solution is the Word Graphs!

Directed & Unweighted Graph

- Nodes → Words
- Edges → join neighbor Words
- Vicinity → Frame N
- Direction → Sequence of words in the original text

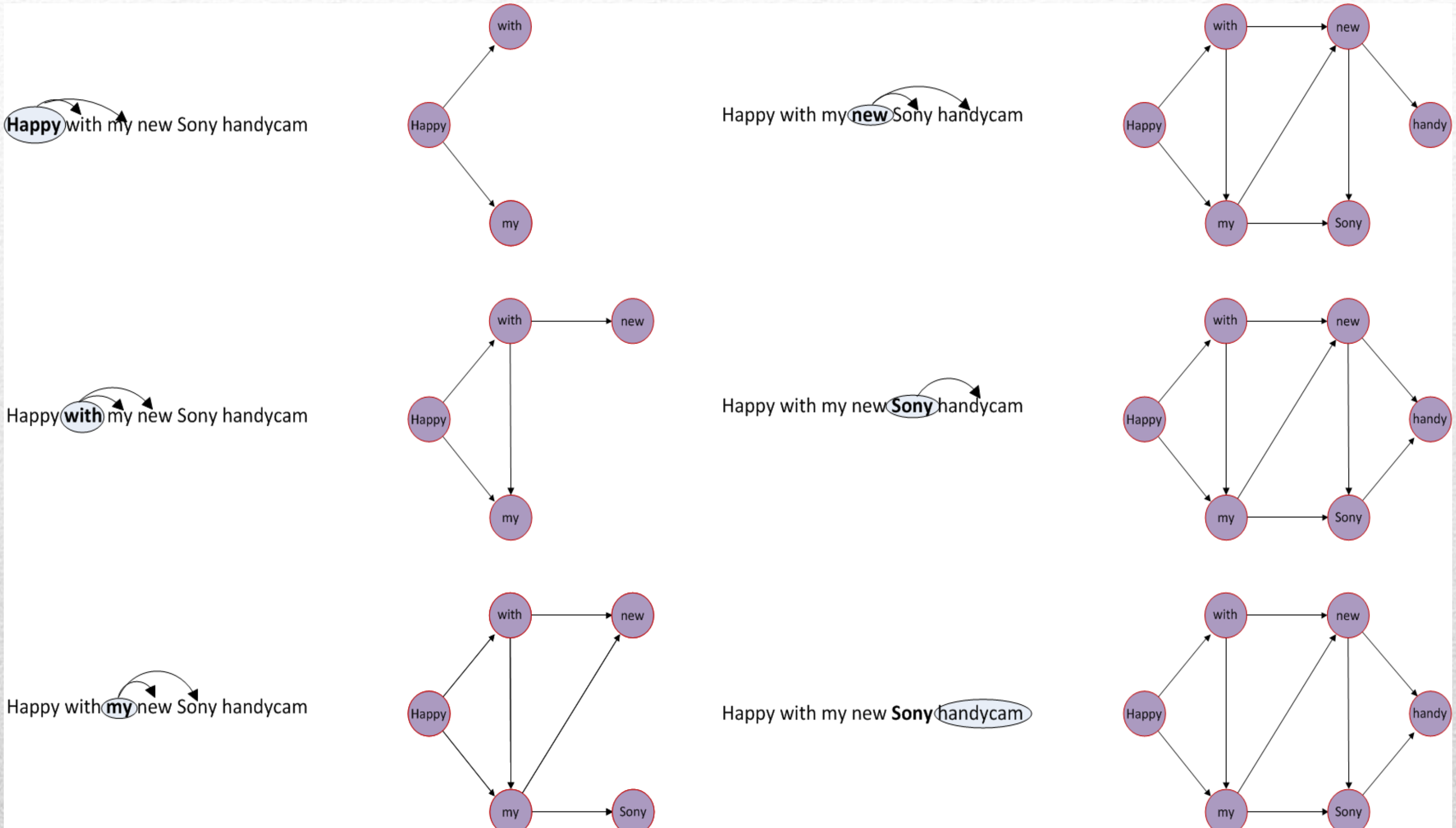
Word Graph Representation

Happy with my new Sony handycam



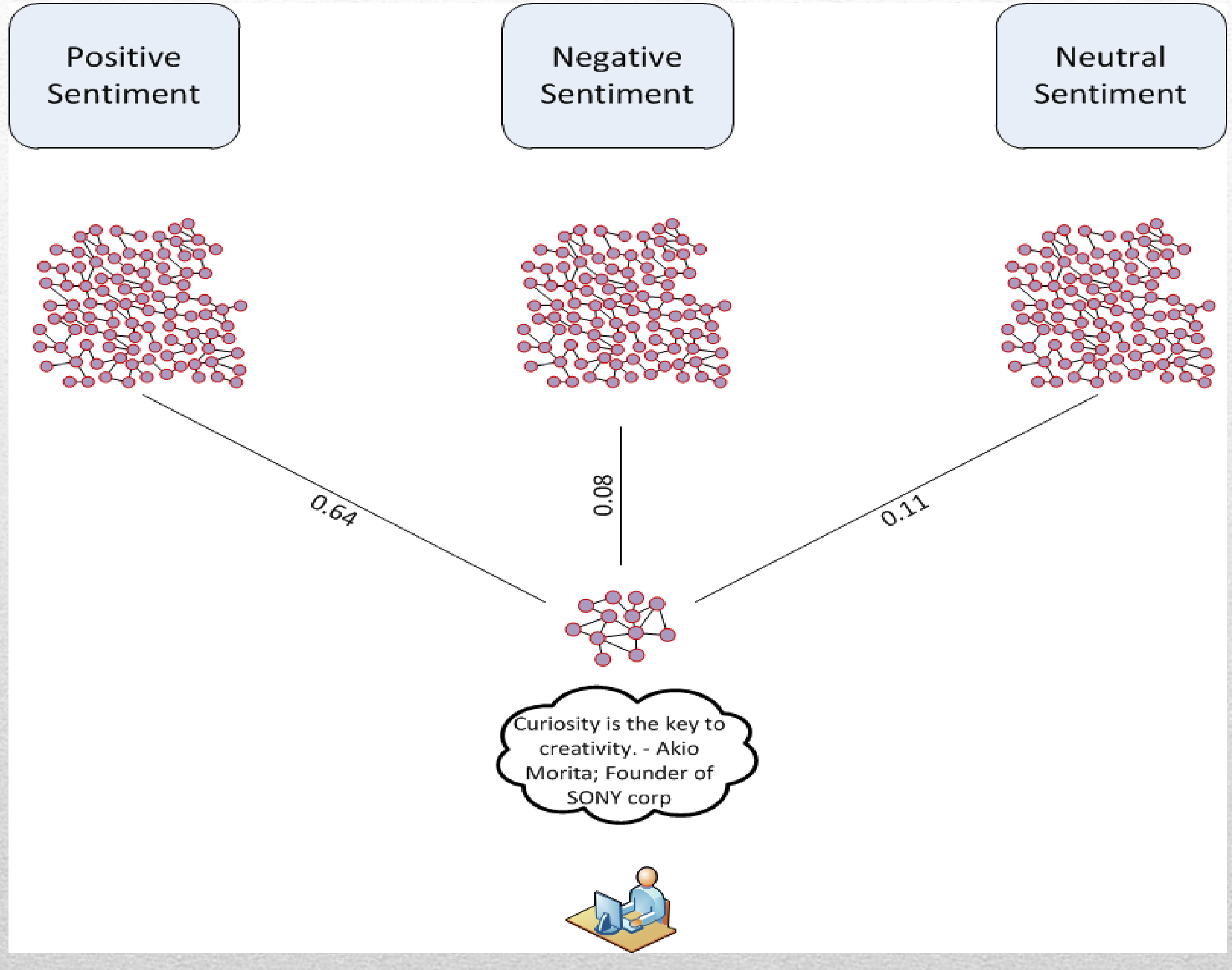
N=2

Graph Construction

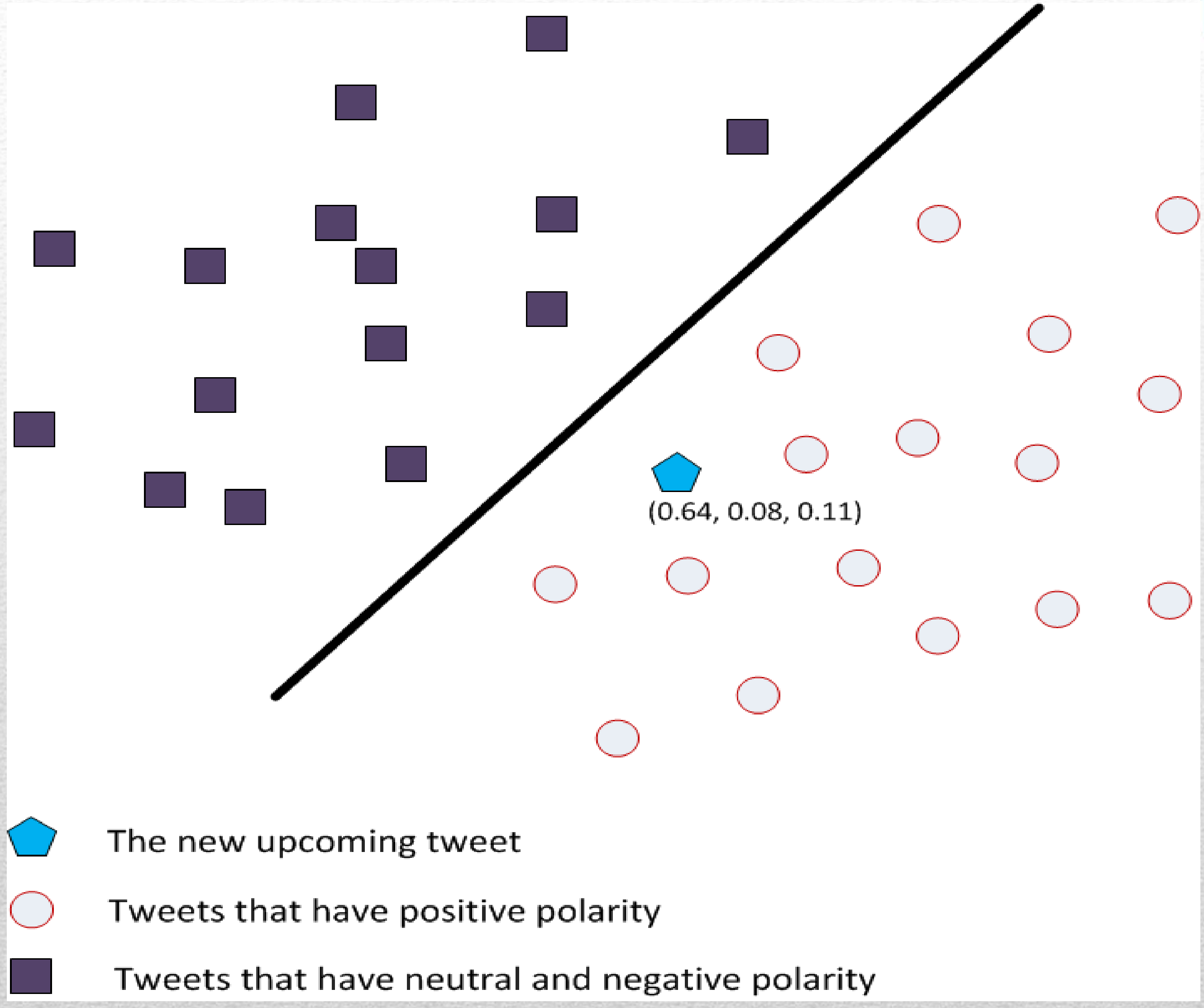


N=2

Word Graph Representation

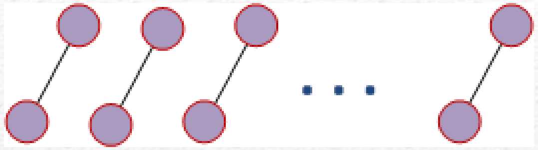


Classification



Graph Similarity Metrics:

- **Containment Similarity**



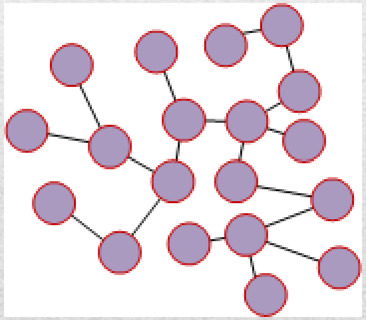
  # Common Edges # Max Edges Normalization

- **Maximum Common SubGraph**

  # Nodes

  # Undirected Edges

  # Directed Edges



ORGANIZATION LOGO

SUPER

Feature Selection techniques:

- Nodes
- Edges

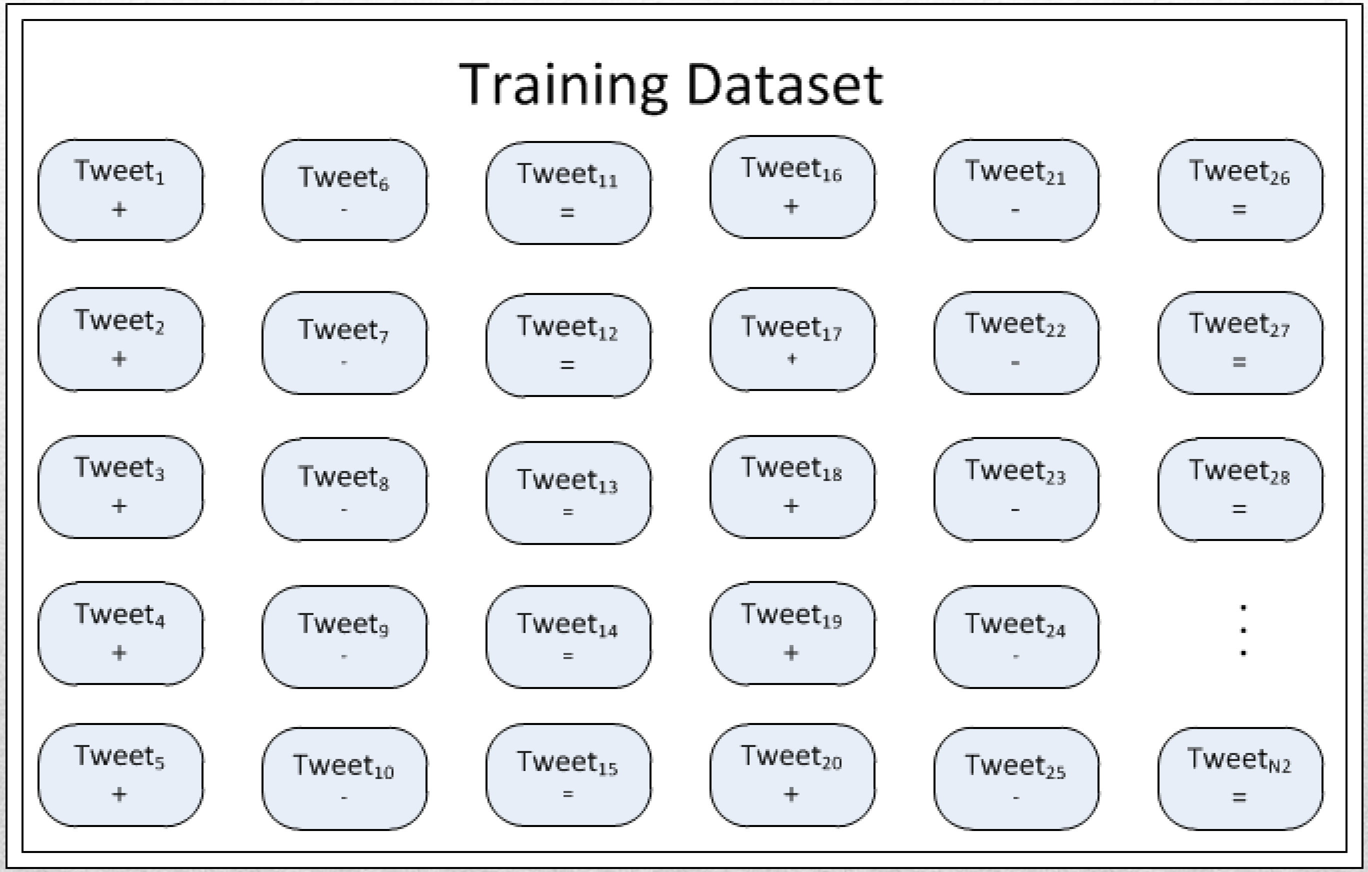
Benefits:

- Improve the Accuracy
- Decrease the Dimensionality & Computation

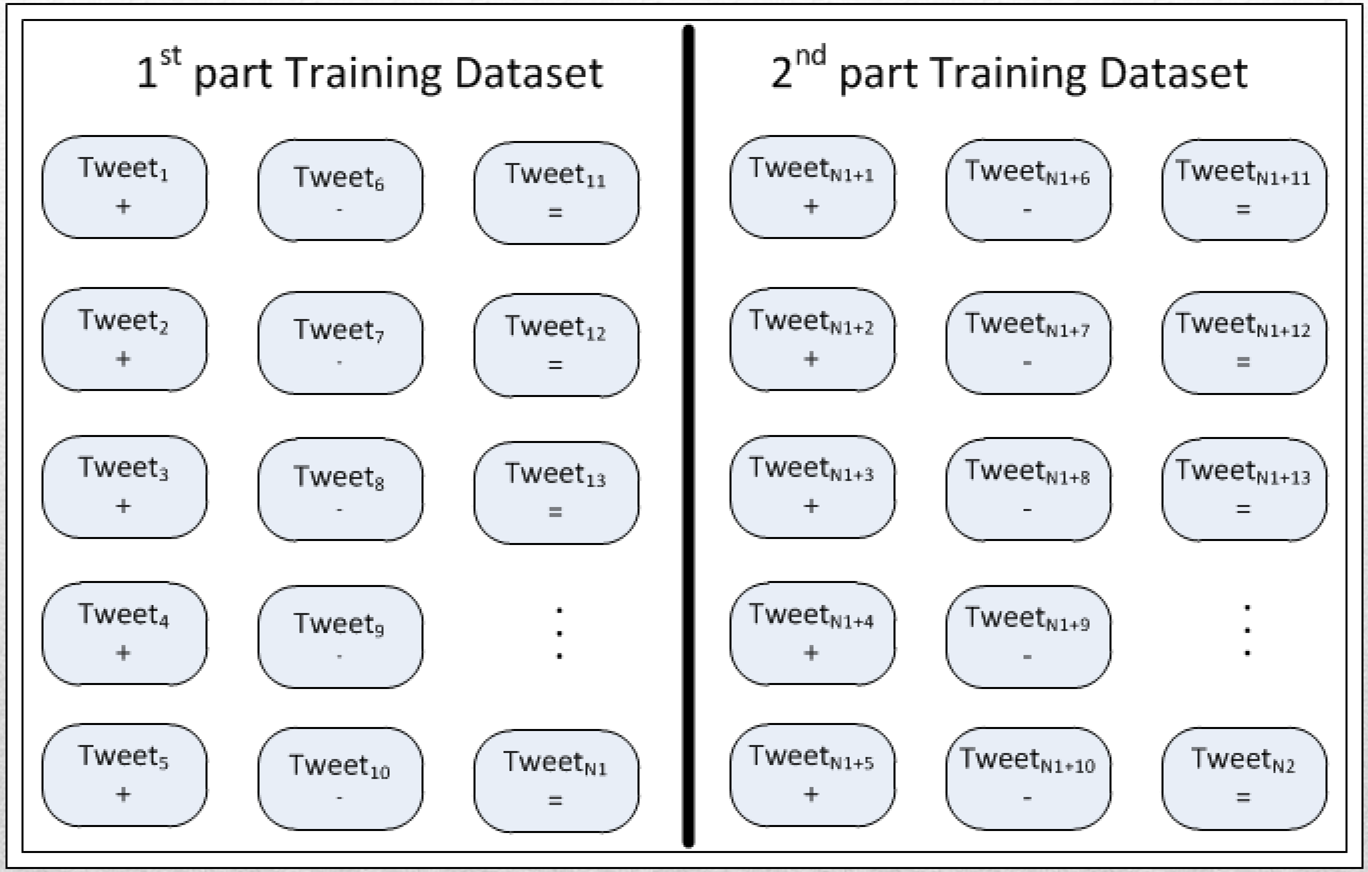
Mutual Information:

The Appropriateness and the Contribution of each Edge for the Accurate Sentiment Prediction + - =

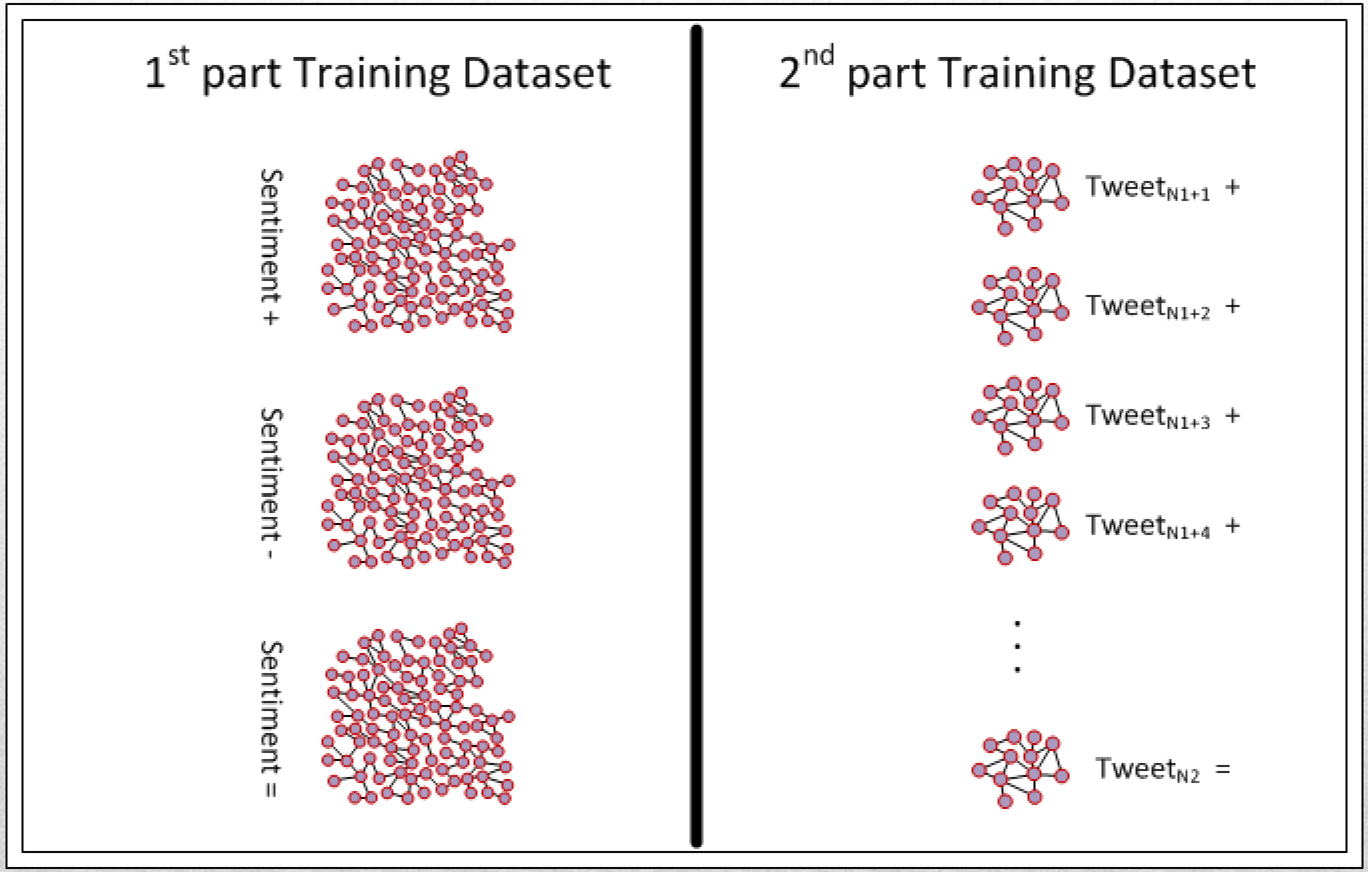
Stage 1/7


Training Dataset
Tweet$_1$ +
Tweet$_2$ +
Tweet$_3$ +
Tweet$_4$ +
Tweet$_5$ +
Tweet$_6$ -
Tweet$_7$ -
Tweet$_8$ -
Tweet$_9$ -
Tweet$_{10}$ -
Tweet$_{11}$ =
Tweet$_{12}$ =
Tweet$_{13}$ =
Tweet$_{14}$ =
Tweet$_{15}$ =
Tweet$_{16}$ +
Tweet$_{17}$ +
Tweet$_{18}$ +
Tweet$_{19}$ +
Tweet$_{20}$ +
Tweet$_{21}$ -
Tweet$_{22}$ -
Tweet$_{23}$ -
Tweet$_{24}$ -
Tweet$_{25}$ -
Tweet$_{26}$ =
Tweet$_{27}$ =
Tweet$_{28}$ =
⋮
Tweet$_{N2}$ =

Stage 2/7

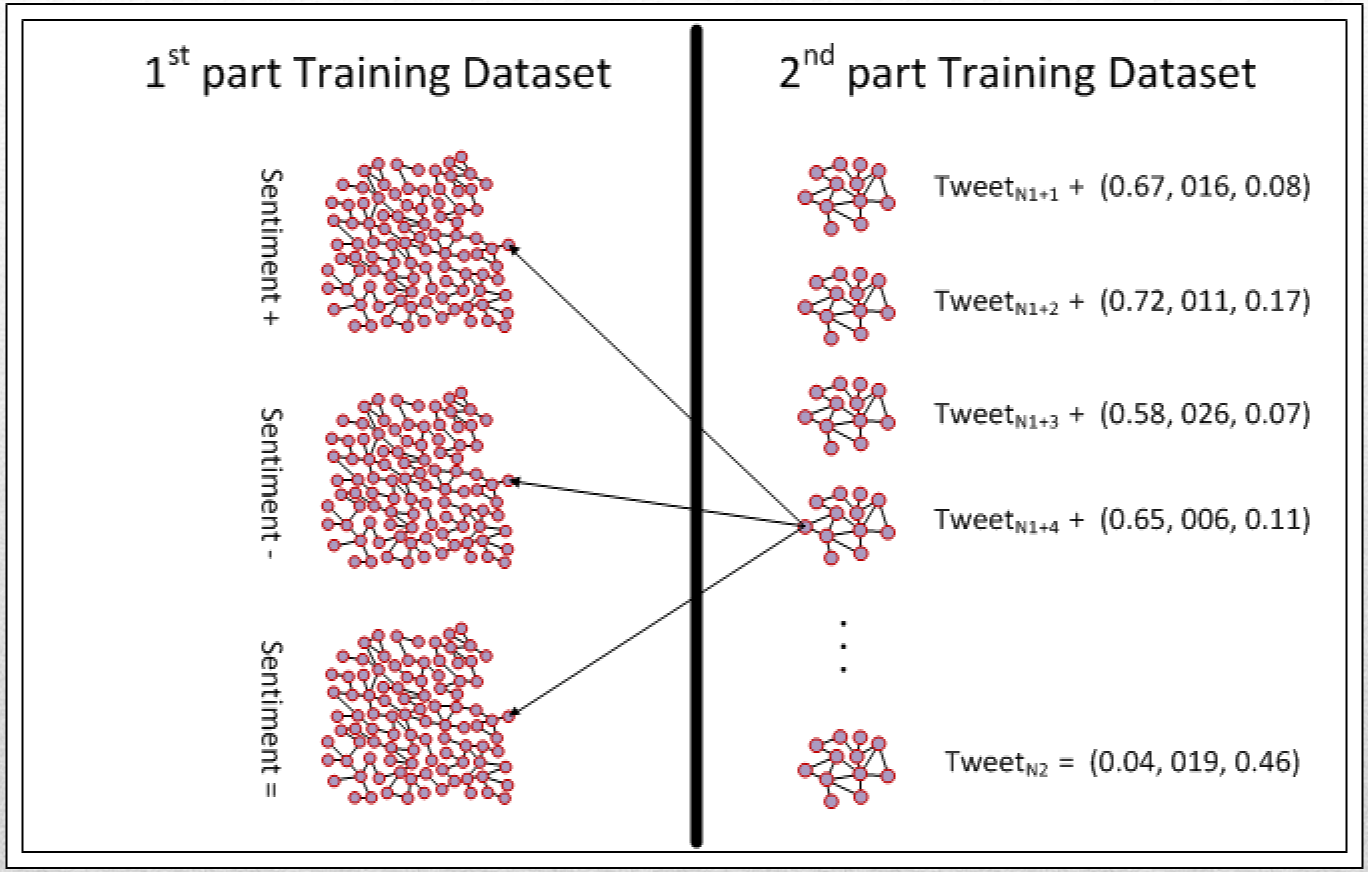

1st part Training Dataset
2nd part Training Dataset
Tweet1 +
Tweet6 -
Tweet11 =
Tweet2 +
Tweet7 -
Tweet12 =
Tweet3 +
Tweet8 -
Tweet13 =
Tweet4 +
Tweet9 -
Tweet5 +
Tweet10 -
TweetN1 =
TweetN1+1 +
TweetN1+6 -
TweetN1+11 =
TweetN1+2 +
TweetN1+7 -
TweetN1+12 =
TweetN1+3 +
TweetN1+8 -
TweetN1+13 =
TweetN1+4 +
TweetN1+9 -
TweetN1+5 +
TweetN1+10 -
TweetN2 =

Stage 3/7



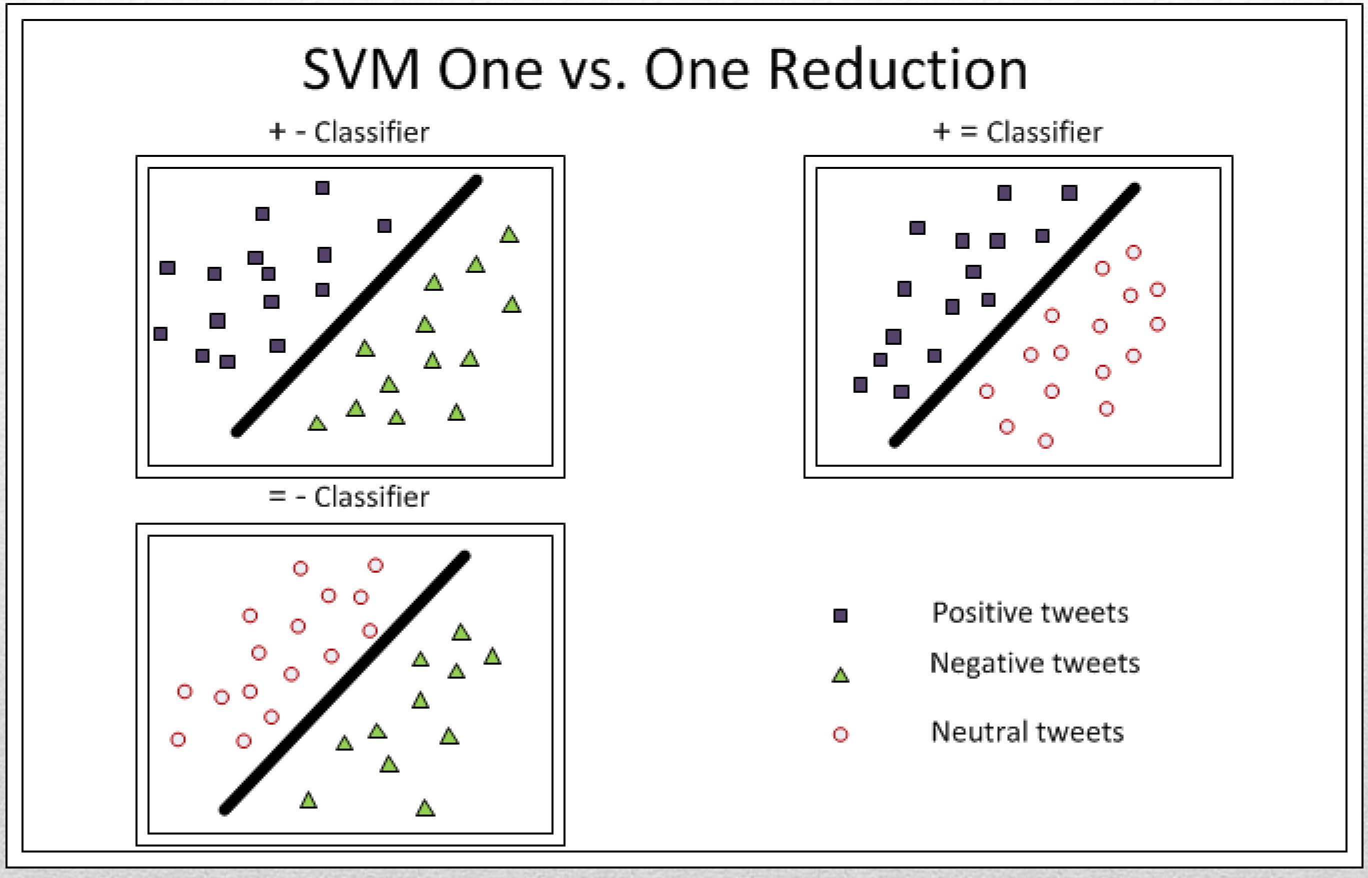
1st part Training Dataset
Sentiment +
Sentiment -
Sentiment =
2nd part Training Dataset
Tweet N1+1 +
Tweet N1+2 +
Tweet N1+3 +
Tweet N1+4 +
Tweet N2 =

Stage 4/7

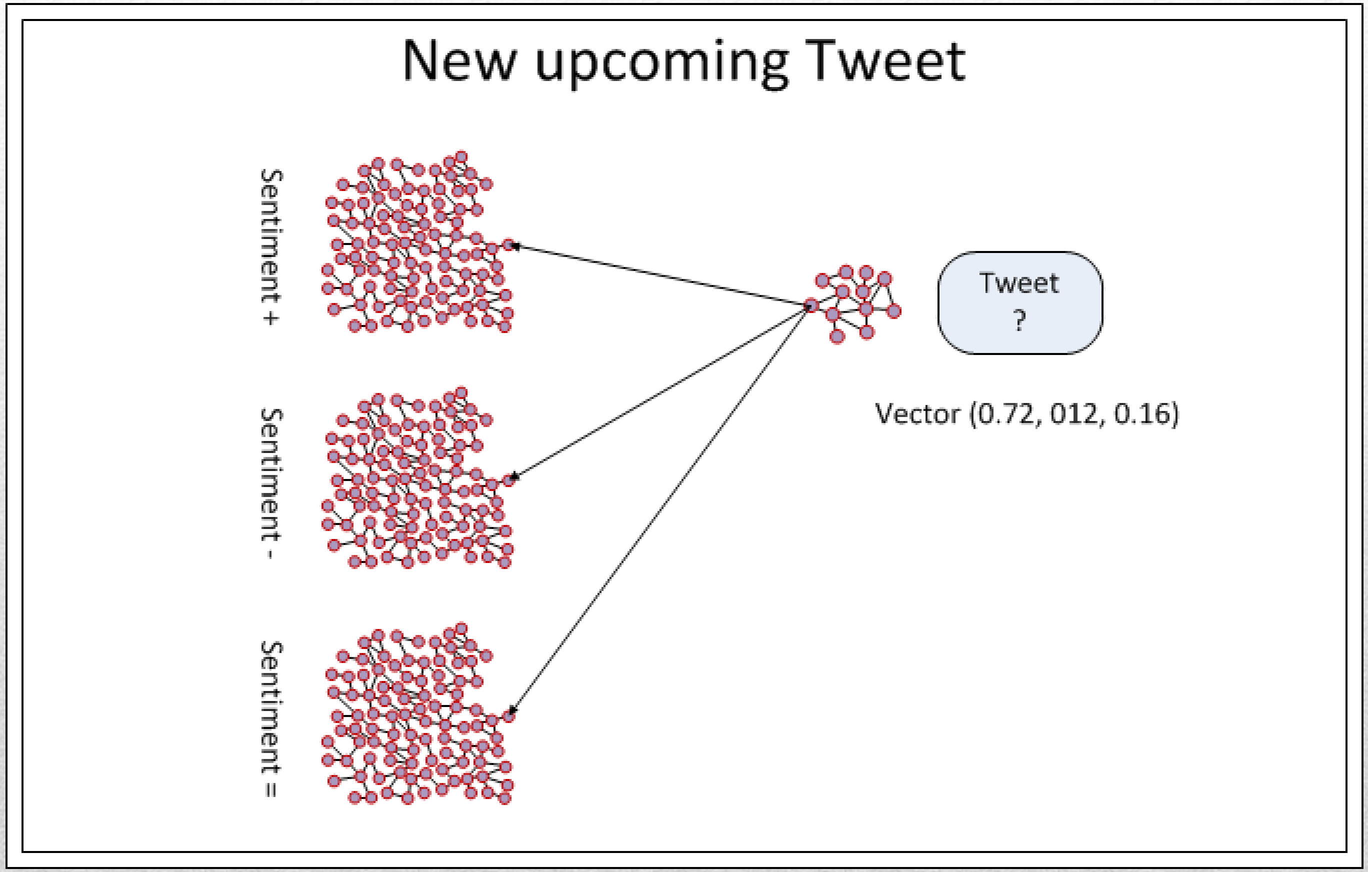


1st part Training Dataset
2nd part Training Dataset
Sentiment +
Sentiment -
Sentiment =
TweetN1+1 + (0.67, 016, 0.08)
TweetN1+2 + (0.72, 011, 0.17)
TweetN1+3 + (0.58, 026, 0.07)
TweetN1+4 + (0.65, 006, 0.11)
TweetN2 = (0.04, 019, 0.46)

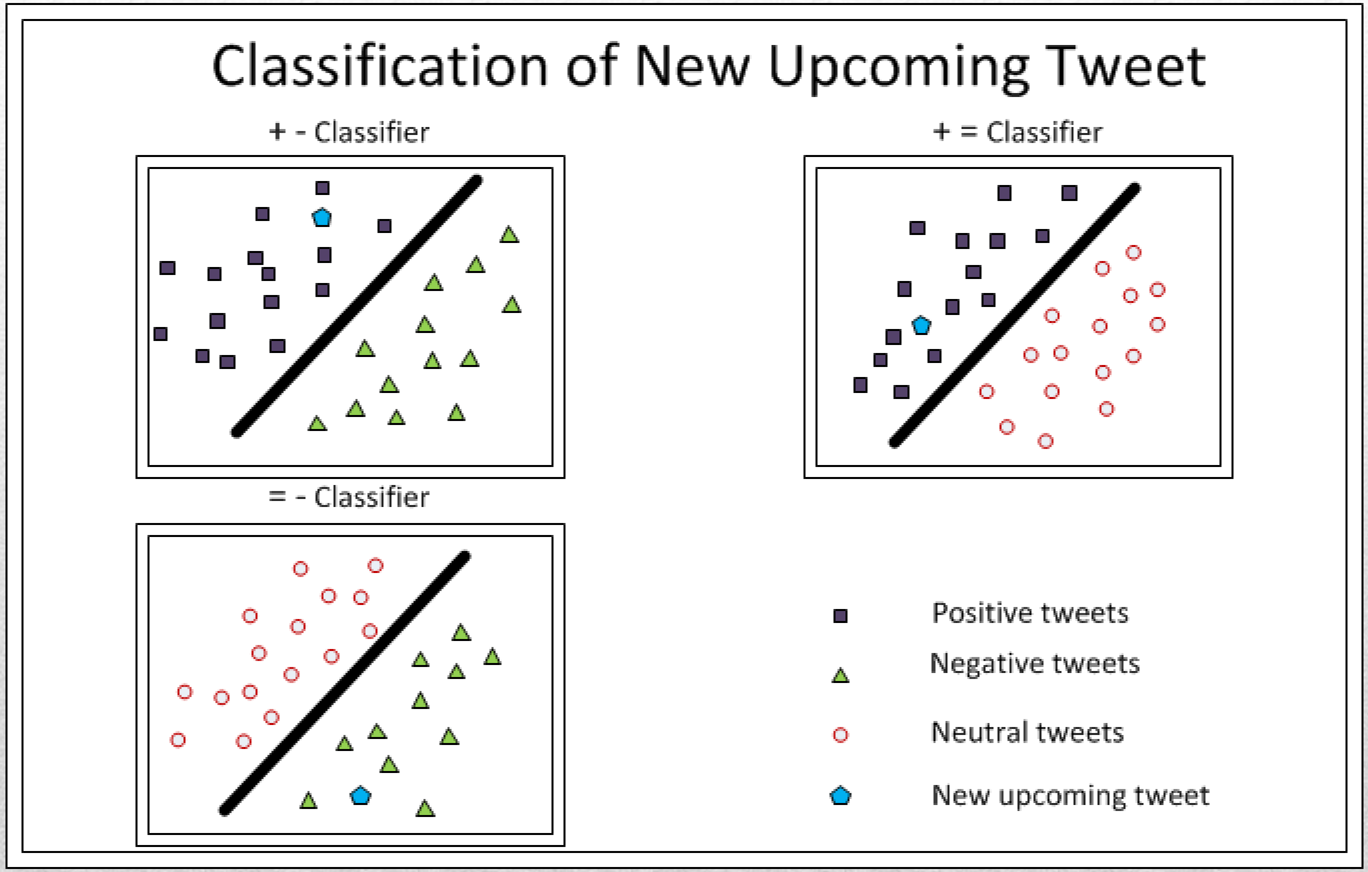
Stage 5/7


SVM One vs. One Reduction
+ - Classifier
+ = Classifier
= - Classifier
Positive tweets
Negative tweets
Neutral tweets

Stage 6/7


New upcoming Tweet
Sentiment +
Sentiment -
Sentiment =
Tweet
?
Vector (0.72, 012, 0.16)

Stage 7/7


Classification of New Upcoming Tweet
+ - Classifier
+ = Classifier
= - Classifier
Positive tweets
Negative tweets
Neutral tweets
New upcoming tweet

Evaluation

Public available Dataset from the research Language-Independent Twitter Sentiment Analysis of Sascha, Hufenhaus and Albayark

The dataset is manually annotated by workers on Amazon's Mehanical Turk

10594 tweets

+ 2334

- 1486

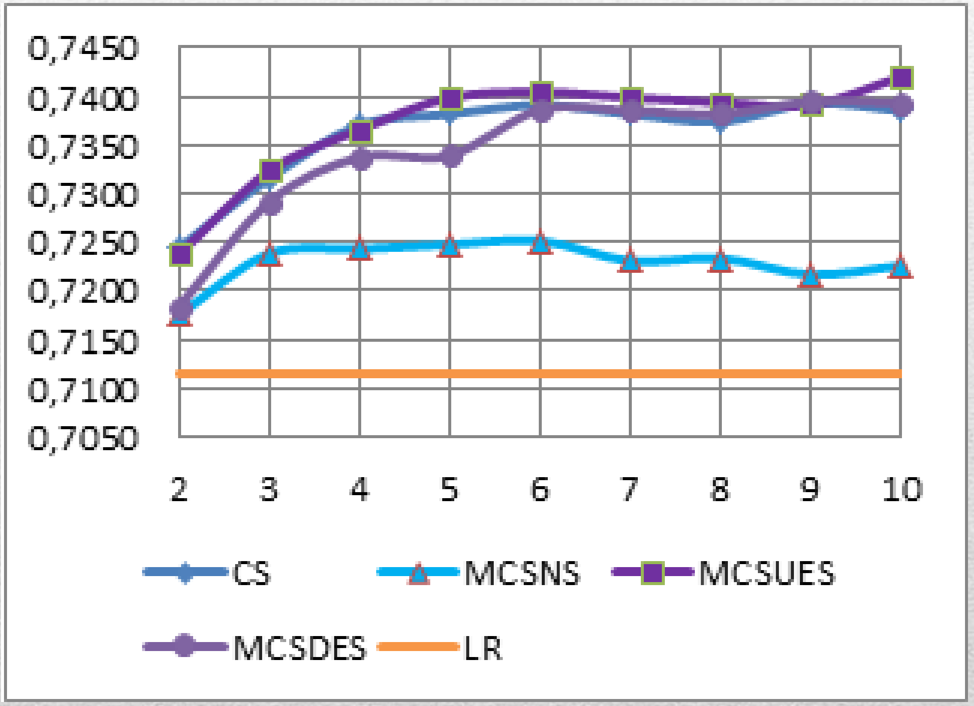
= 6774

SA Accuracy with SOA Methods

Method	4Gram	4Gram Graphs
Bayesian Network	0.6788	0.6791
C4.5	0.6828	0.6896
SVM	0.6777	0.6847
Logistic Regression	0.6822	**0.7115**
Simple Logistic Regression	0.6816	0.7109
Multi-Layer Perceptron	0.6788	0.7069
Best-First Tree	0.6790	0.6840
Functional Tree	0.6822	0.7079

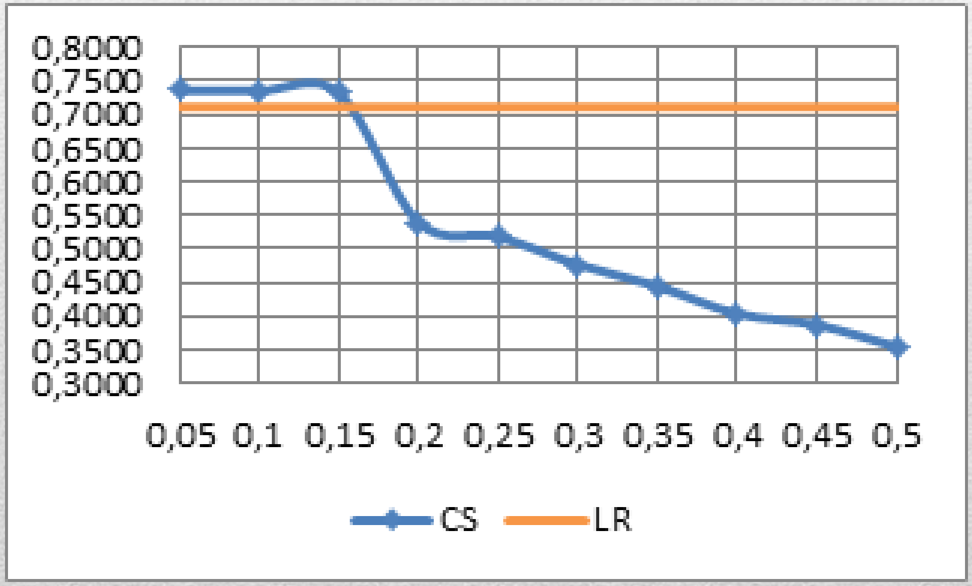
10-Fold Cross Validation
Various size of N (2 - 10) Frame Window
Various Graph Similarity Metrics

No Graph Filering
SVM Classifier



10-Fold Cross Validation
Four-Words Frame size
Mutual Information on Edges as Feature Selection technique
MCS Graph similarity techniques in combination with MI deteriorates the Accuracy of the method

Contanimen Similarity
SVM Classifier

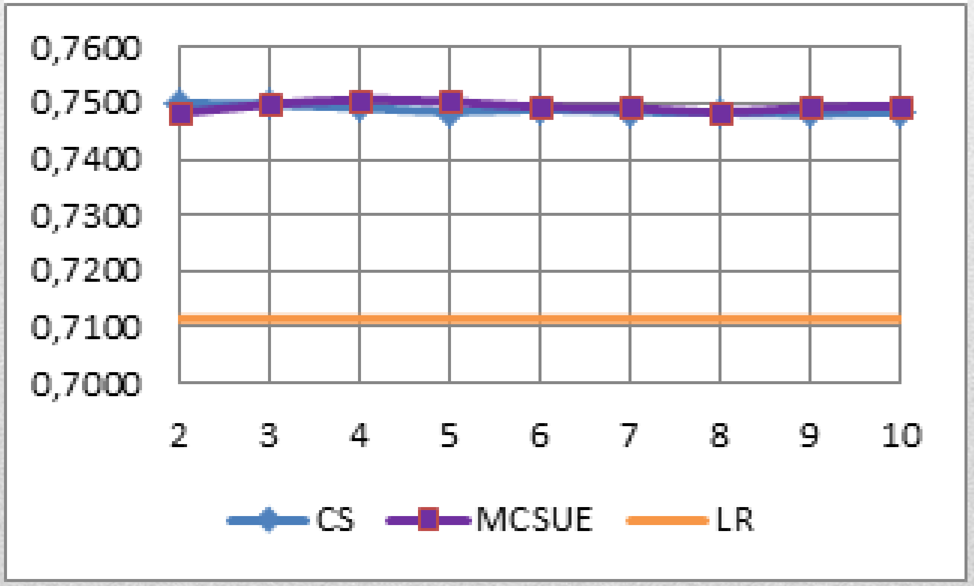


wims'16

SUPER

10-Fold Cross Validation
Various size of N (2 – 10) Frame Window
Containment Similarity
MCSUE Similarity

Gaussian Bayes classifier
No Graph filtering



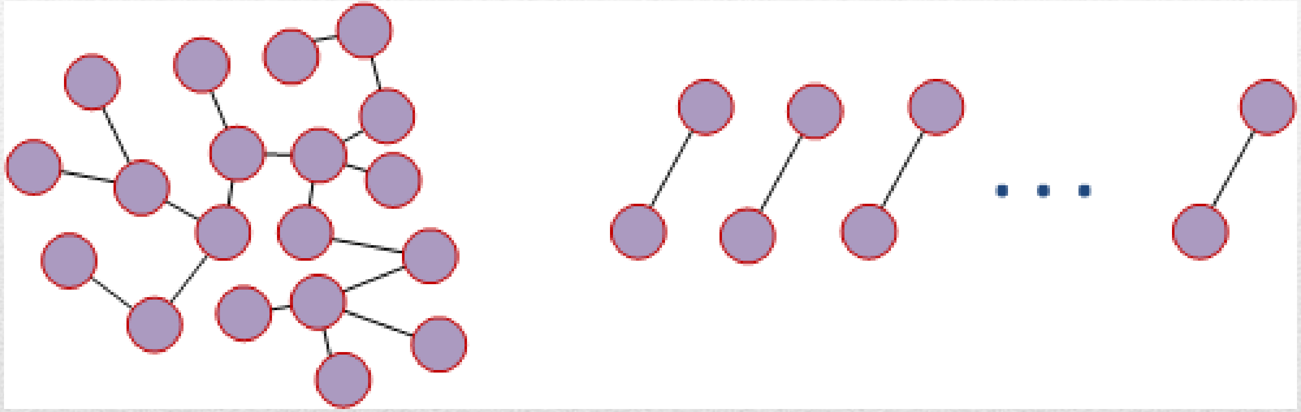
Conclusions 1/3

WGSA:

- Vicinity & sequence of the words-terms.
- Writing characteristics & Partial Matching.
- Polysemy
- Evolution of the Annotated Dataset
- The discard of the edges with low MI has as a result to separate the MCS that captures the sentiment.

Conclusions 2/3

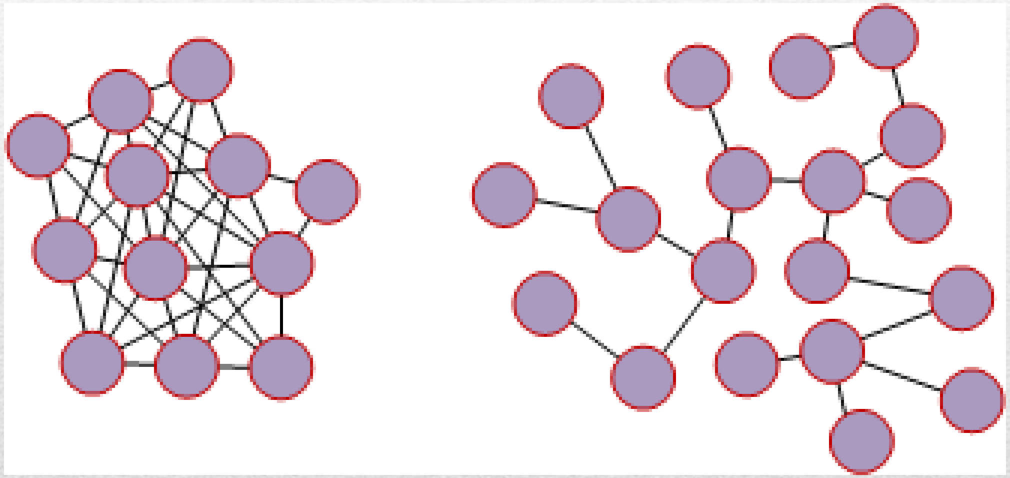
MCS>CS



A correlation among a few common words is more important than the existance of many common word pairs.

Conclusions 3/3

MCSUES>MCSNS



The Dense Correlation among few words is more important than the sparse correlation between many words.

Future work

- Weighted Graphs
- Graph Similarity metrics
- Targeted Sentiment Analysis
- More Sentiments
- Different Training Dataset to Testing Dataset

Thank you for your attention!