Topic Detection and Trend Sensing Via Joint Complexity

Dimitris Milioris[1,2]

[1]Bell Labs, Alcatel-Lucent, France
[2]École Polytechnique ParisTech

Advisor: **Philippe Jacquet**[1,2]

ResCom 2014

Overview

- Motivation & Challenges
- I-Complexity
- Joint Complexity
- Benefits
- Experiments
- Conclusions
- Future work

Motivation

- Online social media services have seen a huge expansion:
  - The value of information has increased dramatically
  - Interactions and communication between users help predict the evolution of information
  - The ability to study Social Networks can provide relevant info in real time

Challenges

The study of Soc. Networks has several research challenges

- Searching in social media is still an open problem
  → short size of posts, tremendous quantity in real time
- Information of the correlation between groups of users
  → predict media consumption, network resources, traffic
  → improve QoS
- Analyze the relationship between members of a group/community
  → reveal important teams
- Spam and adv. detection
  → continuously growing amount of irrelevant info

I – Complexity

- X is a sequence and I(X) is a set of factors (distinct substr.)
- Example: X = apple, then:

  I(X) = {a, p, l, e, ap, pp, pl, le, app, ppl, ple, appl, pple, apple, v}

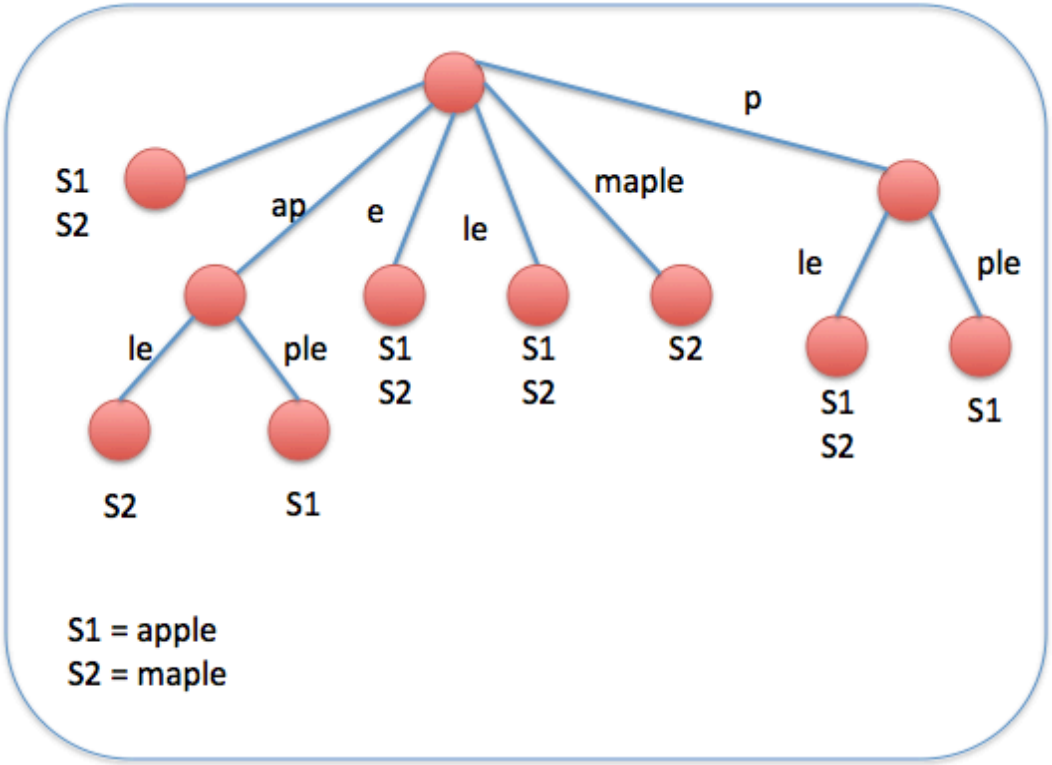
- |I(X)| is the complexity of a sequence
  - |I(X)| = 15 (v denotes the empty string)

Joint Complexity [1]

- The information contained in a string may be revealed by comparing with a reference string
- The Joint Complexity is the number of common distinct factors in two sequences
- $J(X, Y) = |I(X) \cap I(Y)|$
- Efficient way to estimate similarity degree of two sequences
- The analysis of a sequence in subcomponents is done by Suffix Trees
  - Simple, fast and low complexity method to store and recall from memory

[1] P. Jacquet, D. Milioris and W. Szpankowski, “Classification of Markov Sources Through Joint String Complexity: Theory and Experiments”, in IEEE International Symposium on Information Theory (ISIT’13), Istanbul, Turkey, July 2013.

Suffix Trees Superposition [2]

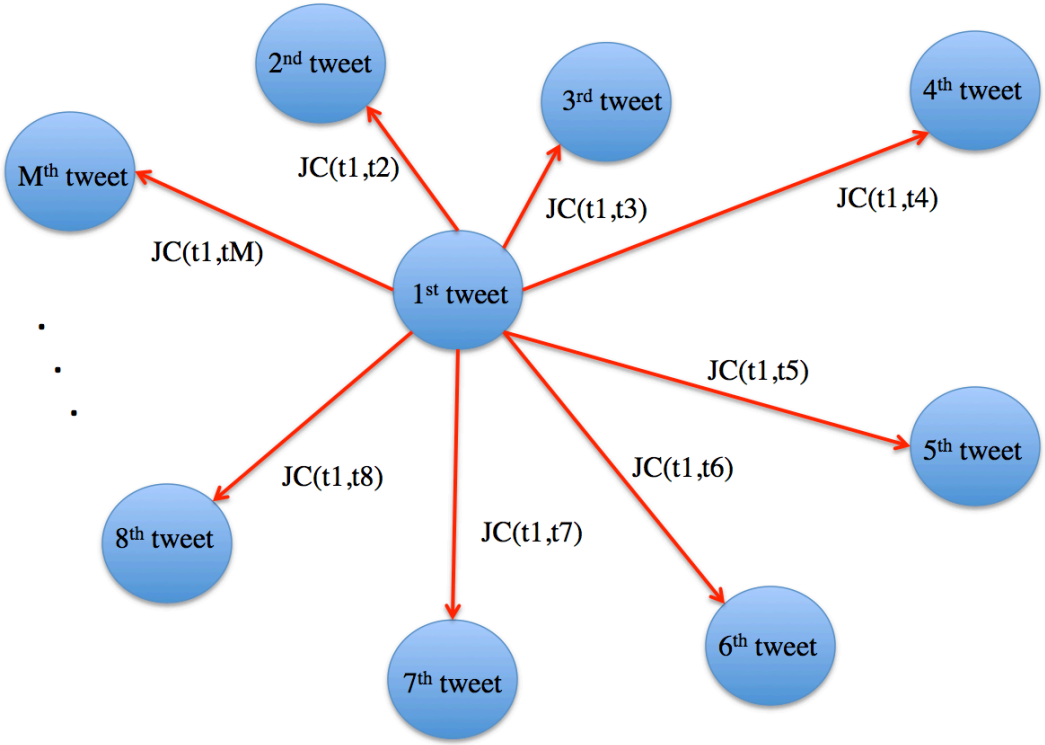


JC(apple, maple) = 9

- Suffix Tree superposition of X = apple and Y = maple
- It reveals the common factors of X and Y, and gives a similarity metric
- Time to build a S.T. = O(n logn)
- Space in memory = O(n), n is the length of the tweet

[2] D. Milioris and P. Jacquet, "Joint Sequence Complexity Analysis: Application to Social Networks Information Flow", in Bell Laboratories Technical Journal, Issue on Data Analytics, Vol. 18, No. 4, 2014

Topic Detection



- Timeslot representation via connected weighted graphs
- Each tweet is a node in the graph and an adjacency matrix (triangular) holds the weight (JC) of every edge

[3] G. Burnside, D. Milioris and P. Jacquet, “One Day in Twitter: Topic Detection via Joint Complexity”, in Snow Data Challenge, Seoul, Korea, April 2014.

Topic Detection

$$T_n = \begin{pmatrix} 0 & JC_{t_1^n,t_2^n} & JC_{t_1^n,t_3^n} & \cdots & JC_{t_1^n,t_M^n} \\ JC_{t_2^n,t_1^n} & 0 & JC_{t_2^n,t_3^n} & \cdots & JC_{t_2^n,t_M^n} \\ JC_{t_3^n,t_1^n} & JC_{t_3^n,t_2^n} & 0 & \cdots & JC_{t_3^n,t_M^n} \\ \vdots & \vdots & & \ddots & \vdots \\ JC_{t_M^n,t_1^n} & JC_{t_M^n,t_2^n} & JC_{t_M^n,t_3^n} & \cdots & 0 \end{pmatrix}$$

$$T_n^{up\ triang} = \begin{pmatrix} \mathbf{JC_{t_1^n,t_2^n}} & \mathbf{JC_{t_1^n,t_3^n}} & \cdots & \mathbf{JC_{t_1^n,t_M^n}} \\ & \mathbf{JC_{t_2^n,t_3^n}} & \cdots & \mathbf{JC_{t_2^n,t_M^n}} \\ & & \ddots & \vdots \\ & & & \mathbf{JC_{t_{M-1}^n,t_M^n}} \end{pmatrix}$$

Algorithms

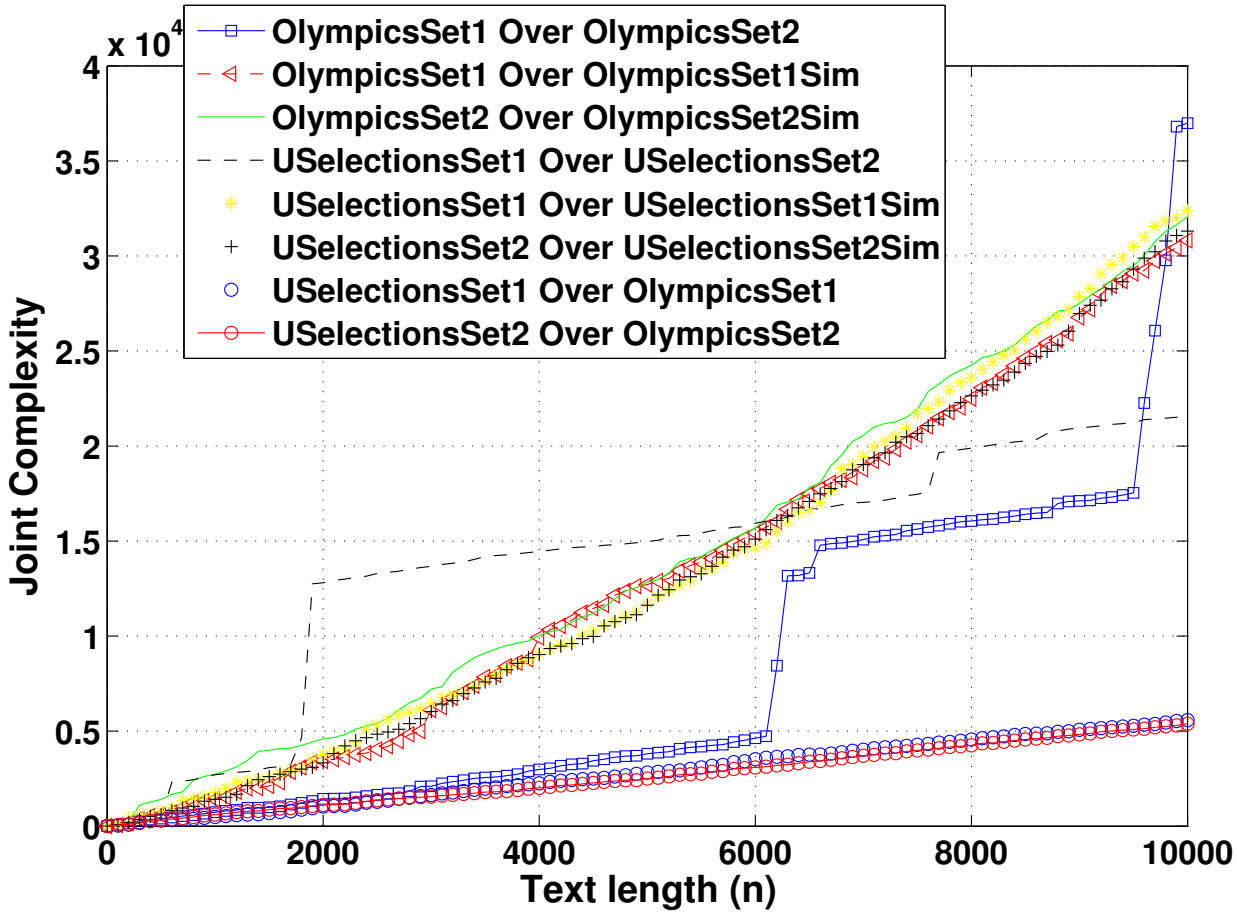
Algorithm 1 Method to retrieve row i of an upper triangular matrix

```
// data is the internal TIntArrayList object
int[] row = new int[data.length+1];
// read the k-th column up to i:
for k = 0 to i − 1 do
    row[k] ← data[k].get(i − k − 1);
end for
row[i] ← 0; // by convention, not computed
if i < data.length − 1 then
    // read the i-th row until the end:
    for j = 0 to data.length − i do
        row[i + j + 1] ← data[i].get(j);
    end for
else do nothing; // the last tweet does not have a row!
end if
return row;
```

Algorithm 2 Topic detection based on Joint Complexity

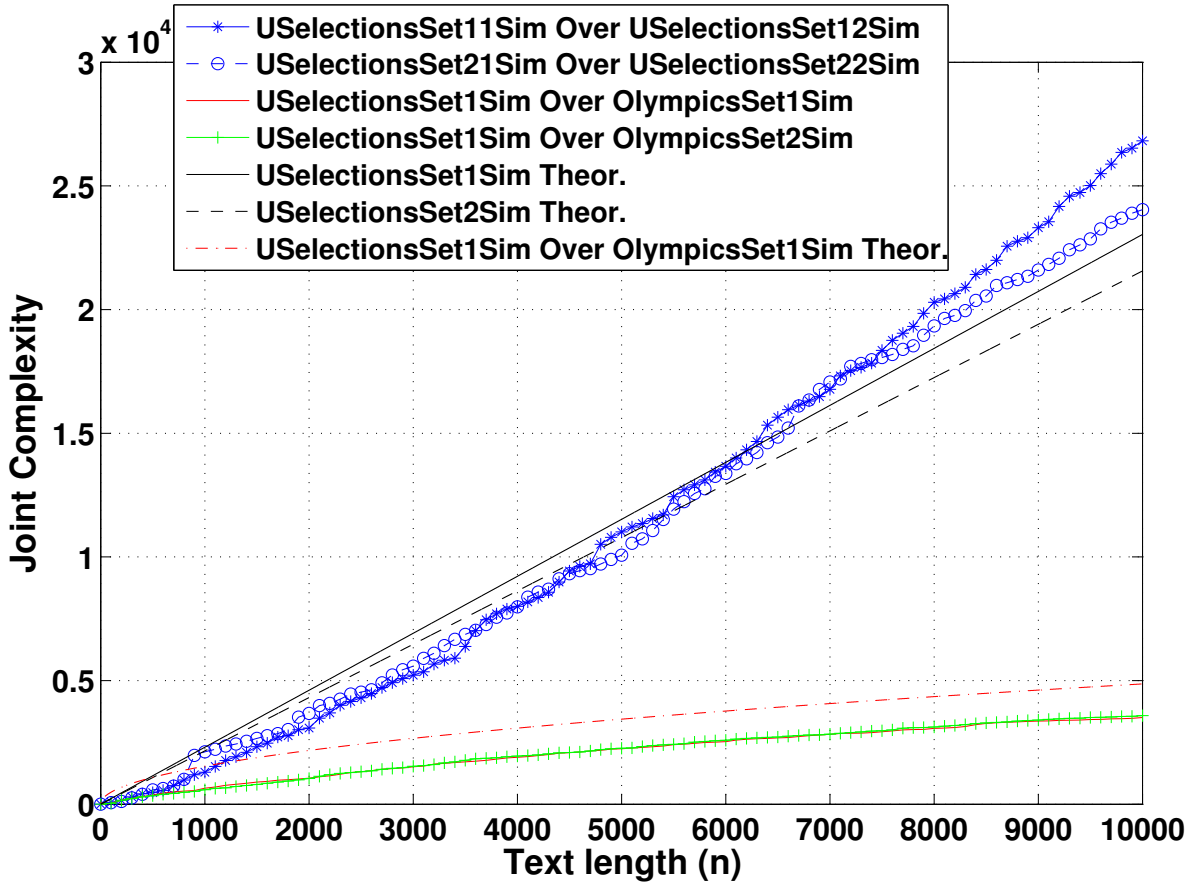
```
// N = # timeslots, M = # tweets in the n-th timeslot
for n = 1 to N do
    for t = 1 to M do
        t ← t_json.getText();
        t_ST ← suffixTreeConstruction(t);
        JCScores ← JCMetric();
    end for
    // Find the most representative & central tweets
    S^n ← sum(JCScores);
    // Get headlines for the central tweets
    R^n ← descendingOrder(S^n);
    // Get set of keywords
    K^n ← keywords(R^n);
    // Get URLs of pictures from the .json file
    P^n ← mediaURL(R^n);
    // Print the results in appropriate format
    Print(R^n);
    Print(K^n);
    Print(P^n);
end for
```

Experiments – Trend Sensing



Joint Complexity of four sets of tweets from the 2012 United States presidential elections and the 2012 Olympic games at London.

Experiments



Joint Complexity of tweet sets from the 2012 United States presidential elections and the 2012 Olympic games at London, in comparison with theoretic curves, using the third Markov order.

Benefits

- Both message classification and identification of the growing trends in real time (trend sensing)
- Track the information and timeline within a social network
- Deal with languages other than English without specific pre-processing or dictionaries, because the method is:
  - simple, context-free, with no grammar and does not use semantics

Conclusions

- Implementation of a topic detection method applied to a dataset of tweets emitted during a 24 hour period
- 3rd Prize in Snow Data Challenge in WWW conf. 2014
- It relies heavily on the concept of Joint String Complexity which has the benefit
  - of being language agnostic and does not require humans to deal with list of keywords
  - has high algorithmic efficiency

Future Work, Improvements

- Use the theoretical background in order to automatically fix the threshold values, than empirical ones (topic detection)

- Extend the JC metric to make topological classification of tweets and perform clustering based on this distance

Publications related to JC

- G. Burnside, D. Milioris and P. Jacquet, “One Day in Twitter: Topic Detection via Joint Complexity”, **Snow Data Challenge, WWW 2014**
- D. Milioris and P. Jacquet, “Joint Sequence Complexity Analysis: Application to Social Networks Information Flow”, in **Bell Laboratories Technical Journal**, Issue on Data Analytics, Vol. 18, No. 4, 2014
- P. Jacquet, D. Milioris, and W. Szpankowski, “Classification of Markov Sources Through Joint String Complexity: Theory and Experiments,” Proc. IEEE Internat. Symp. Inform. Theory (**ISIT ’13**)

- P. Jacquet and W. Szpankowski, “Joint String Complexity for Markov Sources,” Proc. 23rd Internat. Meeting on Probabilistic, Combinatorial, and Asymptotic Methods for the Anal. of Algorithms (**AofA ’12**)
- P. Jacquet, “Common Words Between Two Random Strings,” Proc. IEEE Internat. Symp. on Inform. Theory (**ISIT ’07**)

Publications related to JC

- P. Jacquet and W. Szpankowski, “Analytical Depoissonization and Its Applications,” **Theoret. Comput. Sci**., 201:1-2 (1998), 1–62.

- P. Jacquet and W. Szpankowski, “Autocorrelation on Words and Its Applications: Analysis of Suffix Trees by String-Ruler Approach,” J. **Combin. Theory Ser**. A, 66:2 (1994), 237–269.

Questions ?

dimitrios.milioris@polytechnique.edu