



K-TEAMS / Computer-Supported Collaborative Knowledge Construction Research Laboratory

# User-Level Opinion Propagation Analysis in Discussion Forum Threads

**Dumitru-Clementin Cercel[1] & Ştefan Trăuşan-Matu[1, 2]**

**[1]University POLITEHNICA of Bucharest**

**[2]Romanian Academy Research Institute for Artificial Intelligence**

16th International Conference on Artificial Intelligence: Methodology, Systems, Applications (AIMSA)
Varna, September 11-13, 2014

# Why Opinion Propagation in Online Forum Threads?

- Online forums are very popular among Internet users
- A mixture of positive and negative opinions
- There are multiple applications in the real world

# Opinion Concept

- The definition given in the Oxford Dictionary
  - "A view or judgement formed about something, not necessarily based on fact or knowledge"
  - "A statement of advice by an expert on a professional matter"

- Computational perspective (Ding et al., 2008)
  - Target entity
  - Holder
  - Sentiment
  - Timestamp

X. Ding, B. Liu, and P. S. Yu, "A holistic lexicon-based approach to opinion mining," presented at the Proceedings of the 2008 International Conference on Web Search and Data Mining, Palo Alto, California, USA, 2008.

# Related Work *vs.* Our Original Contribution

- **Related Work**: Drawbacks
  - The Voter Model, The Sznajd Model, The Deffuant Model, The Hegselmann-Krause Model
  - (-) The propagation of opinions in the networks (e.g. the Watts-Strogatz network model or the Barabási-Albert network model)
  - (-) Opinions are numerical values, randomly assigned to individuals
  - (-) The interactions over time between neighboring individuals are also randomly established

- **Our Original Contribution**: Opinion Propagation in Forum Threads
  - Our method consists in determining whether, until a given time step, the users in discussion are in agreement or continue to have different or even contrary opinions.
    - (+) We take into account the opinions written by users
    - (+) We use the real-world's online forum threads

# Online Forum Thread

- $(T_{DT}, S_{DT}, U_{DT}(t_\tau), P_{DT}(t_\tau), R_{DT}(t_\tau))$ at time step $t_\tau$, $\tau \in \mathbb{N}^*$

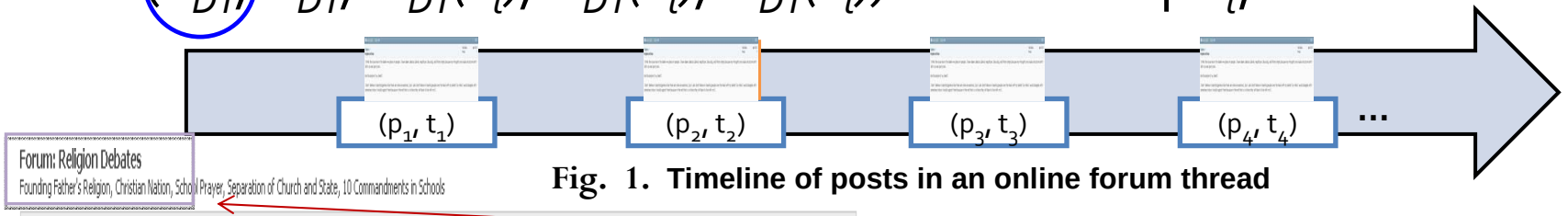



**Fig. 1. Timeline of posts in an online forum thread**


Topic

# Online Forum Thread

- $(T_{DT}, S_{DT}, U_{DT}(t_\tau), P_{DT}(t_\tau), R_{DT}(t_\tau))$ at time step $t_\tau$, $\tau \in \mathbb{N}^*$

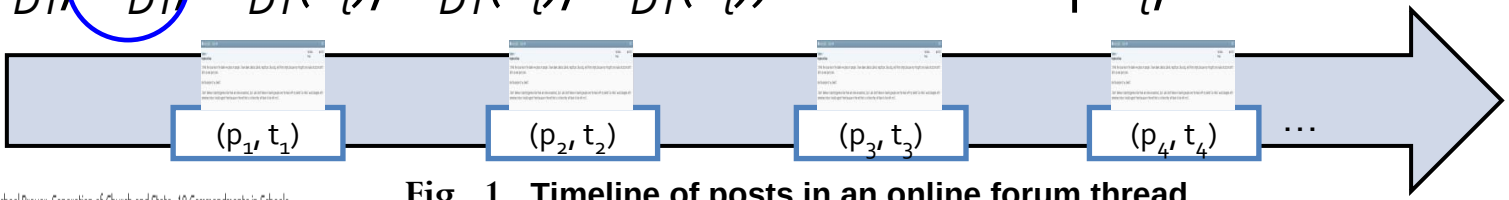
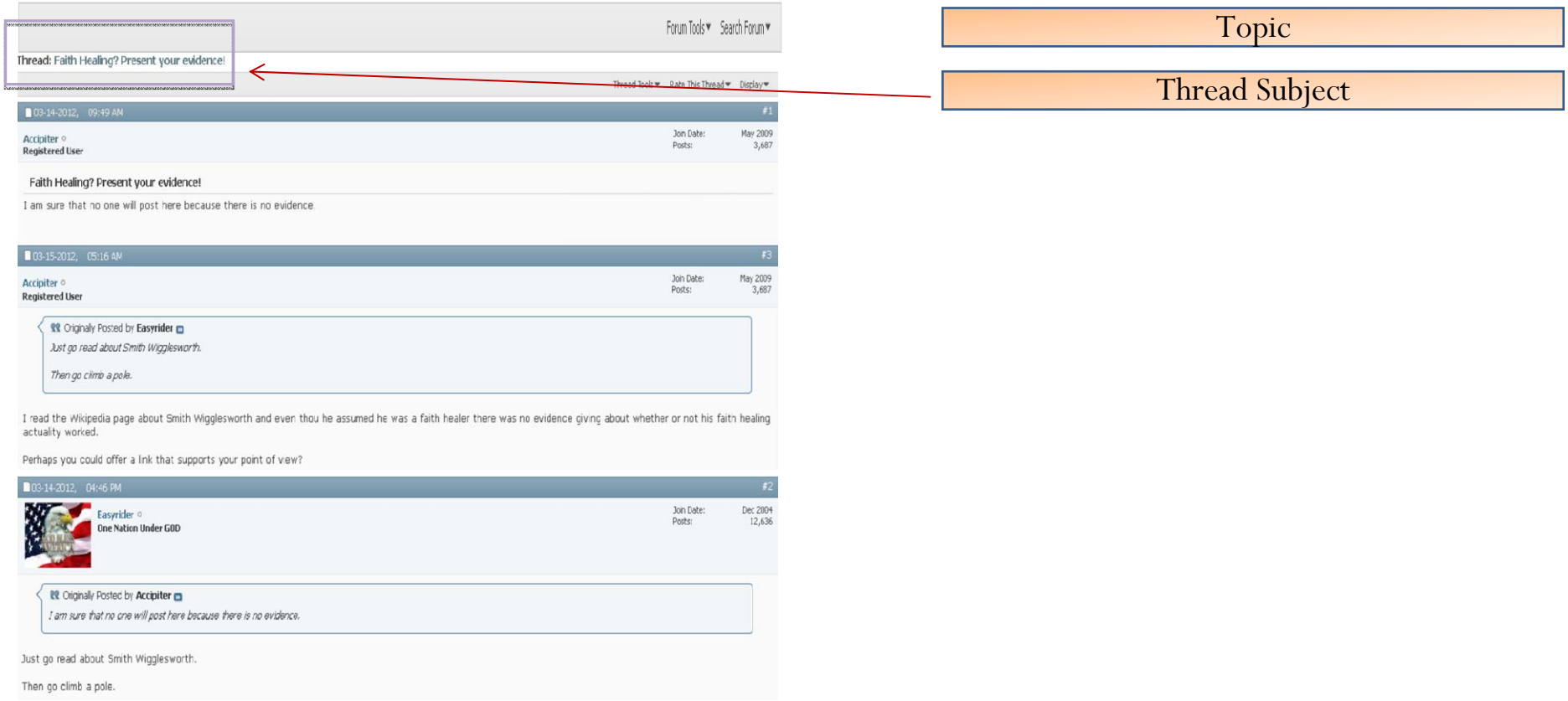



**Fig. 1. Timeline of posts in an online forum thread**



# Online Forum Thread

- $(T_{DT}, S_{DT}, U_{DT}(t_\tau), P_{DT}(t_\tau), R_{DT}(t_\tau))$ at time step $t_\tau$, $\tau \in \mathbb{N}^*$

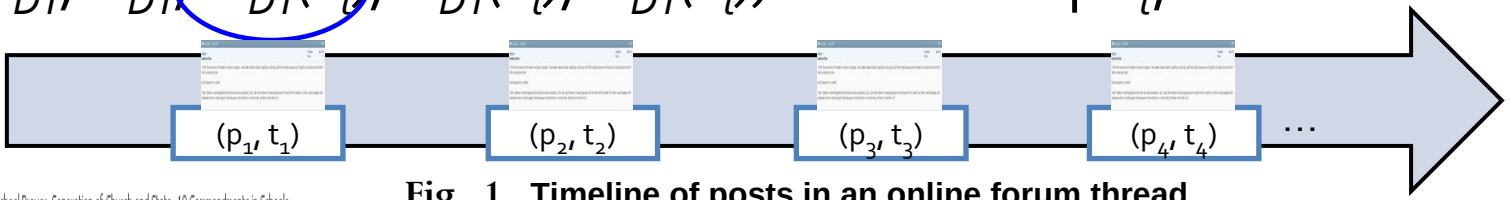
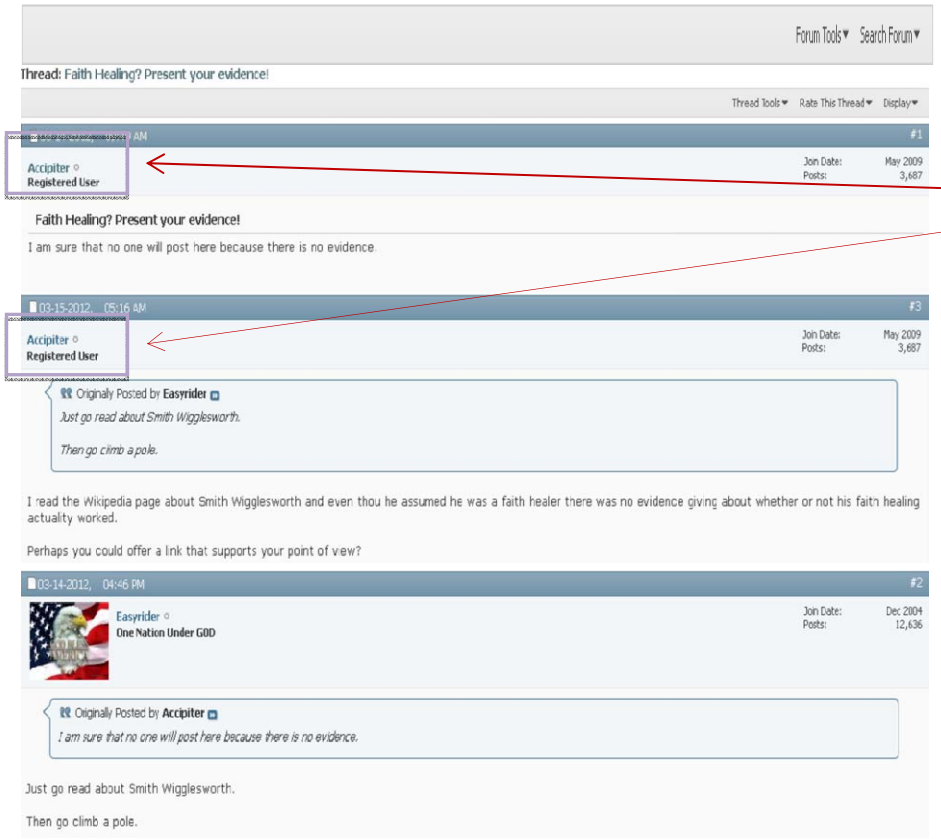



**Fig. 1. Timeline of posts in an online forum thread**




Topic

Thread Subject

A set of users

# Online Forum Thread

- $(T_{DT}, S_{DT}, U_{DT}(t_\tau), P_{DT}(t_\tau), R_{DT}(t_\tau))$ at time step $t_\tau$, $\tau \in \mathbb{N}^*$

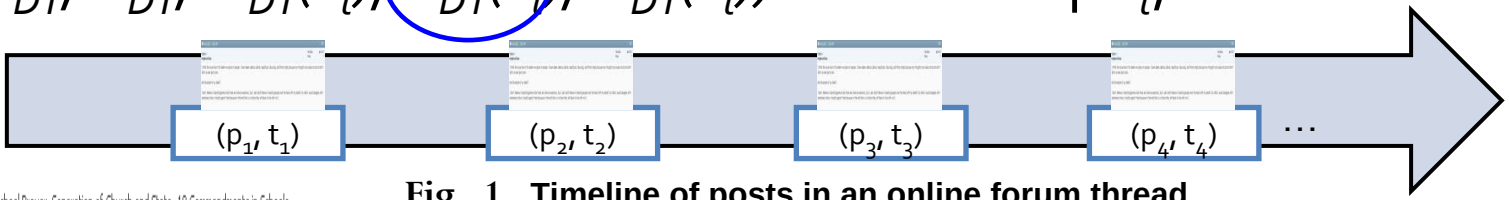
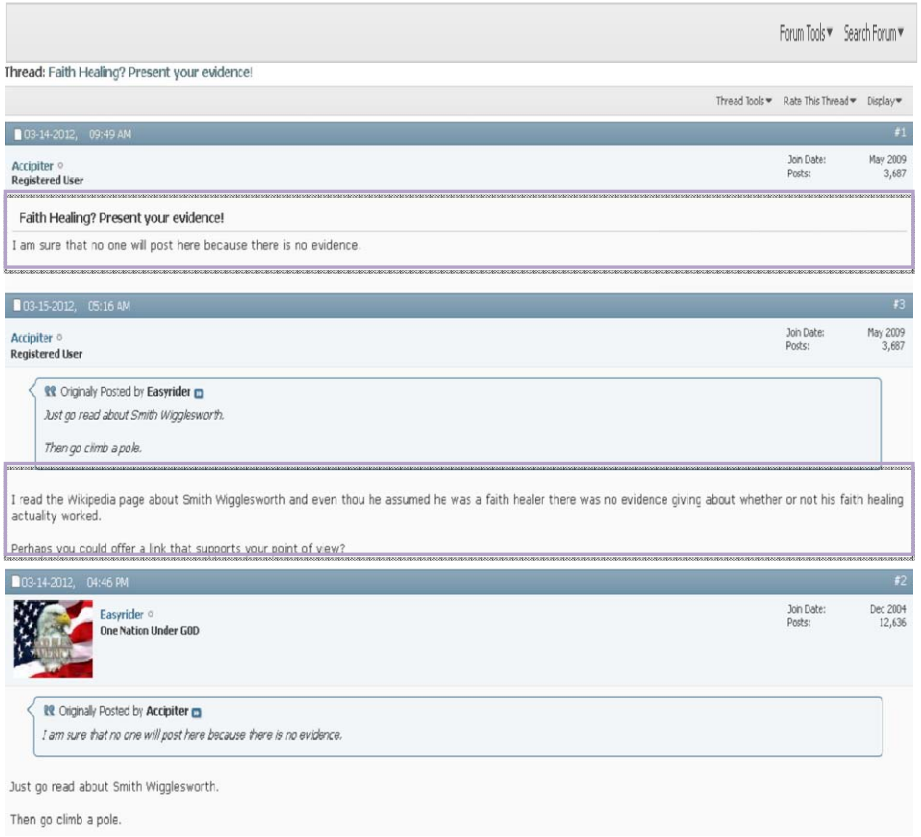



**Fig. 1. Timeline of posts in an online forum thread**




Topic

Thread Subject

A set of users

A set of posts, so that $p \in P_{DT}(t_\tau)$ represents interventions of the users $u \in U_{DT}(t_\tau)$ on the subject $S_{DT}$ at time step $t_\tau$

# Online Forum Thread

- $(T_{DT}, S_{DT}, U_{DT}(t_\tau), P_{DT}(t_\tau), R_{DT}(t_\tau))$ at time step $t_\tau$, $\tau \in \mathbb{N}^*$

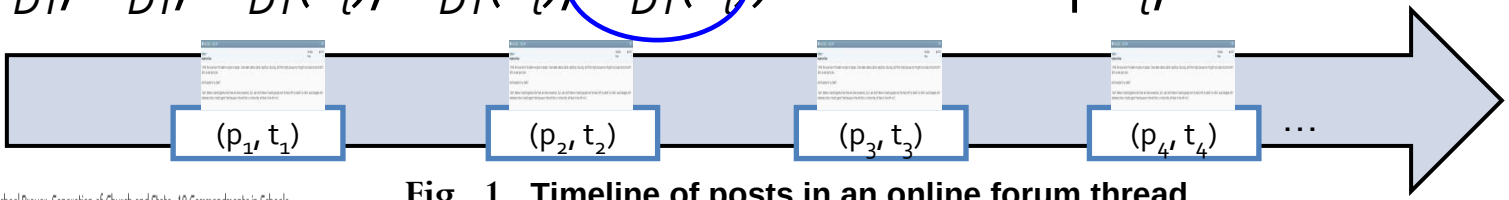
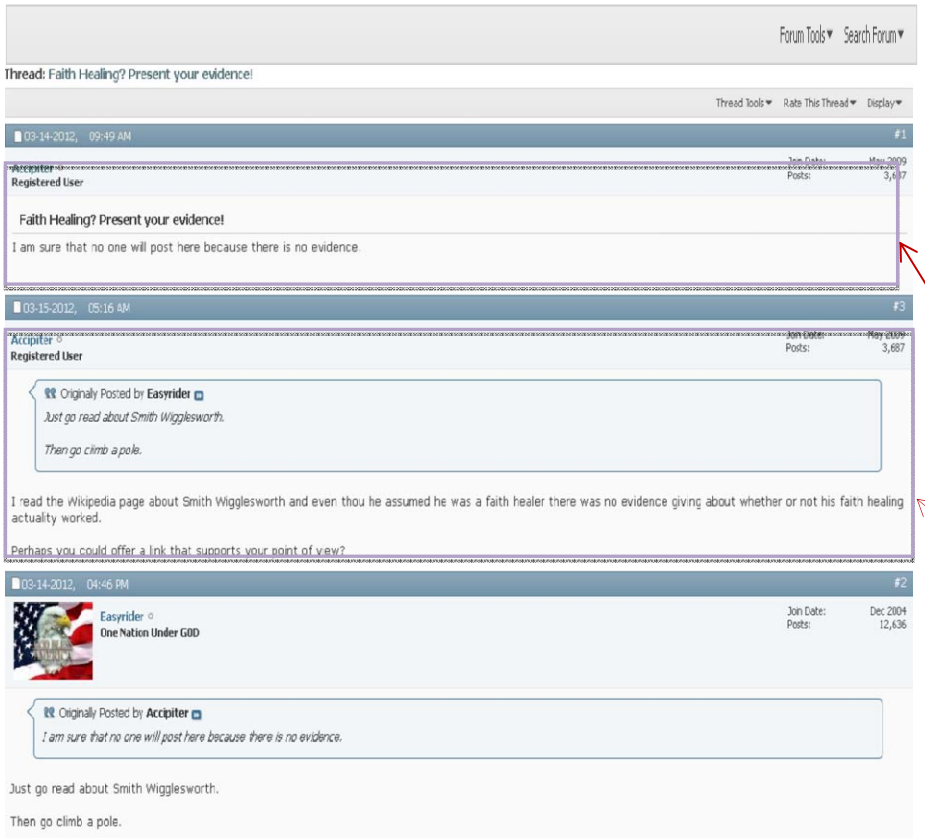



**Fig. 1. Timeline of posts in an online forum thread**




Topic

Thread Subject

A set of users

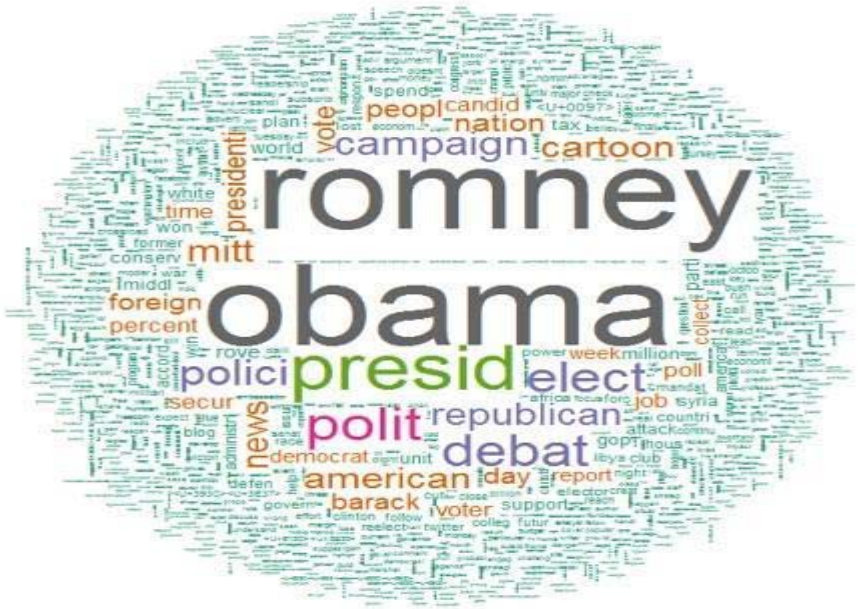
A set of posts, so that $p \in P_{DT}(t_\tau)$ represents interventions of the users $u \in U_{DT}(t_\tau)$ on the subject $S_{DT}$ at time step $t_\tau$

$R_{PP}(t_\tau) \subset P_{DT}(t_\tau) \times P_{DT}(t_\tau)$

- *belong-to relation* $R_{UP}(t_\tau) \subset U_{DT}(t_\tau) \times P_{DT}(t_\tau)$, so that, if $(u, p) \in R_{UP}(t_\tau)$, the post $p \in P_{DT}(t_\tau)$ was written by the user $u \in U_{DT}(t_\tau)$.
- *reply-to relation* $R_{PP}(t_\tau) \subset P_{DT}(t_\tau) \times P_{DT}(t_\tau)$, so that, if $(p_1, p_2) \in R_{PP}(t_\tau)$, the post $p_1 \in P_{DT}(t_\tau)$ replies to another post $p_2 \in P_{DT}(t_\tau)$.

# Concepts

- Noun Term Vocabulary
  - All distinct noun terms on which the users $U_{DT}(t_\tau)$ expressed their opinions and that are semantically related to one of the noun terms by which the subject $S_{DT}$ is written
  - $V^d_{DT}(t_\tau) = \{n_1, n_2, \ldots, n_d\}$ denotes a $d$-dimensional vocabulary of noun terms at time step $t_\tau$, $\tau \in \mathbb{N}^*$



# Concepts

- Discrete Opinion Space
  - The opinions of each user $u \in U_{DT}(t_\tau)$ on the noun terms from the vocabulary $V^d_{DT}(t_\tau)$ can be represented by a vector in a $d$-dimensional discrete opinion space $OS^d_{DT} = \{-1, 0, +1\}^d$
  - $o^d_{DT}(t_\tau) = [o_1\ o_2\ \ldots\ o_d]^T$ denotes an opinion vector at time step $t_\tau$, $\tau \in \mathbb{N}^*$, in the $d$-dimensional discrete opinion space $OS^d_{DT}$
    - The opinion entries $o_k$ can take one of the following sentiment scores: -1, 0, or +1

- Observation!
  - If the user $u \in U_{DT}(t_\tau)$ does not express his opinion on the noun term $s_k \in V^d_{DT}(t_\tau)$ until time step $t_\tau$, $\tau \in \mathbb{N}^*$, then we consider the value 0 for the entry $o_k \in o^d_{DT}(t_\tau)$
  - If until time step $t_\tau$, $\tau \in \mathbb{N}^*$, the user $u \in U_{DT}(t_\tau)$ gives more opinions on a noun term, then only his last opinion is taken into consideration

# Concepts

- Term-User Opinion Matrix
  - We construct a $d \times n$ term-user matrix $A_{T\text{-}U}(t_\tau) = [A_1(t_\tau)\ A_2(t_\tau)\ \dots\ A_n(t_\tau)]$ at time step $t_\tau$, $\tau \in \mathbb{N}^*$
    - $n$ denotes the number of users in the set $U_{DT}(t_\tau)$
    - Each column $A_u(t_\tau) = [a_{1,u}(t_\tau)\ a_{2,u}(t_\tau)\ \dots\ a_{d,u}(t_\tau)]^T$ corresponds to a user $u \in U_{DT}(t_\tau)$ and denotes the $d$-dimensional opinion vector of the user $u \in U_{DT}(t_\tau)$ in the discrete opinion space $OS^d_{DT}$
- User-User Similarity Matrix
  - We construct a $n \times n$ user-user similarity matrix $B_{U\text{-}U}(t_\tau)$ at time step $t_\tau$, $\tau \in \mathbb{N}^*$
  - The entry of the row $k^{th}$ and of the column $h^{th}$ of the matrix $B_{U\text{-}U}(t_\tau)$ is denoted by $b_{k,h}(t_\tau)$ and represents the similarity between users $k$ and $u \in U_{DT}(t_\tau)$ from the perspective of the opinion vector expressed by these users
  - $b_{k,h}(t_\tau) = sim(A_k(t_\tau) = [a_{1,k}(t_\tau)\ a_{2,k}(t_\tau)\ \dots\ a_{d,k}(t_\tau)]^T, A_h(t_\tau) = [a_{1,h}(t_\tau)\ a_{2,h}(t_\tau)\ \dots\ a_{d,h}(t_\tau)]^T) =$

$$\frac{\sum_{i=1}^{d} a_{i,k}(t_\tau)\, a_{i,h}(t_\tau)}{\sqrt{\sum_{i=1}^{d} \left(a_{i,k}(t_\tau)\right)^2}\ \sqrt{\sum_{i=1}^{d} \left(a_{i,h}(t_\tau)\right)^2}}$$

# Problem Formalization

- The Problem of User-Level Opinion Propagation in Online Forum Threads:
  - ***Input***: *Given, at time step* $t_\tau$, $\tau \in \mathbb{N}^*$, *a subset of users* $U'_{DT}(t_\tau) \subset U_{DT}(t_\tau)$ *who have similar opinion vectors for any time step* $t_i$, $t_i \geq t_\tau$, *(i.e.* $b_{u2u1} \leq \varepsilon_1$, $\square\ u_2, u_1 \in U'_{DT}(t_\tau)$*) and who initiated the opinion propagation process at time step* $t_\tau$
  - ***Goal***: *A user* $u \in U_{DT}(t_j) \setminus U'_{DT}(t_\tau)$ *is considered to be influenced, at time step* $t_j$, $t_j > t_\tau$ *by the opinion propagation only if the following condition is met:*

$$b_{u3,u}(t_\tau) \leq b_{u1,u}(t_j) + \varepsilon_2$$

*where* $u_3 \in U_{DT}(t_j) \setminus U'_{DT}(t_\tau)$, $\square\ u_1 \in U'_{DT}(t_\tau)$ *and the parameters* $\varepsilon_1$, $\varepsilon_2$ *can be* [illegible] *set.*

The opinion vector of the user $u$ should be dissimilar to the opinion vectors of the users $U_{DT}(t_j) \setminus U'_{DT}(t_\tau)$

The opinion vector of the user $u$ should be similar to the opinion vectors of the users $U'_{DT}(t_\tau)$

We study the case in which $t_\tau = t_1$ and the set $U'_{DT}(t_1) = \{u_1\}$

# Proposed System

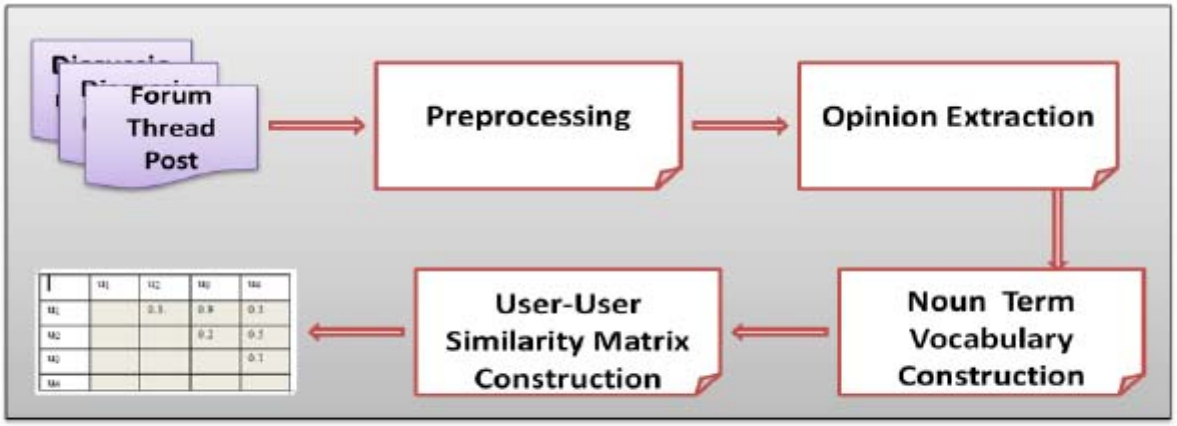
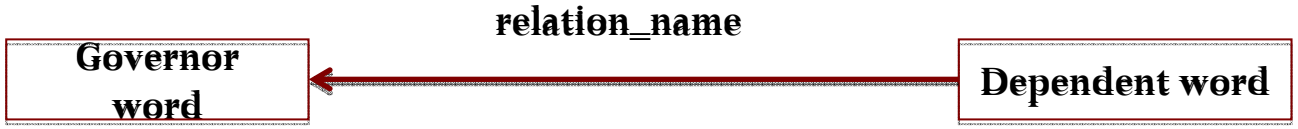



**Fig.** 2. General architecture of the user-level opinion propagation analysis in online forum threads

- Step 1: Preprocessing
  - tokenization, part-of-speech tagging, syntactic parsing, and coreference resolution

# Proposed System

- Step 2: Opinion Extraction
  - The Stanford Dependencies:
    - binary semantic relations in a sentence between a *governor* and a *dependent* (abbreviated_relation_name(governor, dependent))




    - "dobj", "nsubj", "amod", "acomp", "advmod", "xcomp ", "neg"
    - For example, "nsubj" (nominal subject): the governor is any adjective, and the dependent is a noun term
      - For the sentence " The movie is interesting" the Stanford dependencies are: det (movie, The), nsubj (interesting, movie), cop ( interesting, is), root (ROOT, interesting)

# Proposed System

- Step 3: Noun Term Vocabulary Construction

**Algorithm 1:** Construction of Noun-Term Vocabulary

**Input:** $P_{DT}(t_\tau) = \{p_\tau\}_{\tau \in \mathbb{N}}$ – set of posts in the forum thread at time step $t_\tau$;
**Input:** $S_{DT} = \{w_k\}_{k \in \mathbb{N}}$ – the subject of the forum thread;
**Output:** $V^d_{DT}(t_\tau)$ – noun-term vocabulary at time step $t_\tau$;

1: $\Omega \leftarrow$ MiningDependencyRelations $(P_{DT}(t_\tau))$ // $\Omega = \{(h_m, d_n)\}_{m,n \in \mathbb{N}}$
2: **for each** word $w_k$ in $S_{DT}$ **do**
3: **if** (**checkNoun**($w_k$) and length($w_k$) > 3) **then**
4: $w_k \leftarrow$ lemmatization($w_k$)
5: $w_k \leftarrow$ lowercase($w_k$)
6: **else**
7: $S_{DT} \leftarrow S_{DT} \setminus w_k$
8: **end if**
9: **end for**
10: **for each** pair $(h_m, d_n)$ in $\Omega$ **do**
11: sim $\leftarrow \sum_{w_k \in S_{DT}} sim_{wu}(h_m, w_k)$
12: **if** sim != 0 and $h_m \notin V^d_{DT}(t_\tau)$ **then**
13: $V^d_{DT}(t_\tau) \leftarrow V^d_{DT}(t_\tau) \cup (h_m)$
14: **end if**
15: **end for**

**Algorithm 2:** Mining Dependency Relations from a Forum Thread

**Input:** $P_{DT}(t_\tau) = \{p_i\}_{i \in \mathbb{N}}$ – set of posts in the forum thread at time step $t_\tau$;
**Output:** $\Omega = \{(h_m, d_n)\}_{m,n \in \mathbb{N}}$ – pairs (noun_term, opinion_word)

$\Omega \leftarrow \emptyset$
**for each** post $p_i$ in $P_{DT}(t_\tau)$ **do**
$R(p_i) \leftarrow$ Parse($p_i$) // the set $\{rel_{i,j}(h_{i,k}, d_{i,l})\}_{j,k,l \in \mathbb{N}}$ of dependency relations from post $p_i$;
**for each** dependency relation $rel_{i,j}(h_{i,k}, d_{i,l})$ in $R(p_i)$ **do**
**if** ($rel_{i,j}$ = 'dobj' and **checkVerb**($h_{i,k}$) and **checkNoun**($d_{i,l}$)) **then**
$\Omega \leftarrow \Omega \cup (h_{i,k}, d_{i,l})$
**end if**
**if** ($rel_{i,j}$ = 'nsubj' and **checkAdjective**($h_{i,k}$) and **checkNoun**($d_{i,l}$)) **then**
$\Omega \leftarrow \Omega \cup (h_{i,k}, d_{i,l})$
**end if**
**if** ($rel_{i,j}$ = 'amod' and **checkAdjective**($h_{i,l}$) and **checkNoun**($d_{i,k}$)) **then**
$\Omega \leftarrow \Omega \cup (d_{i,l}, h_{i,k})$
**end if**
**if** ($rel_{i,j}$ = 'advmod' and **checkVerb**($h_{i,k}$) and there is $rel_{i,jj}(h_{i,kk}, d_{i,ll})$ so that $rel_{i,jj}$ = 'nsubj' and $h_{i,k} = h_{i,kk}$ **then**
$\Omega \leftarrow \Omega \cup (d_{i,l}, d_{i,ll})$
**end if**
**if** ($rel_{i,j}$ = 'acomp' and **checkVerb**($h_{i,k}$) and there is $rel_{i,jj}(h_{i,kk}, d_{i,ll})$ so that $rel_{i,jj}$ = 'rcmod' and $h_{i,k} = d_{i,ll}$ **then**
$\Omega \leftarrow \Omega \cup (d_{i,l}, h_{i,kk})$
**end if**
**if** ($rel_{i,j}$ = 'xcomp' and **checkVerb**($h_{i,k}$) and there is $rel_{i,jj}(h_{i,kk}, d_{i,ll})$ so that $rel_{i,jj}$ = 'ccomp' and $h_{i,k} = d_{i,ll}$ **then**
$\Omega \leftarrow \Omega \cup (h_{i,kk}, d_{i,l})$
**end if**
**end for**
**end for**

# Proposed System

- Step 3: Noun Term Vocabulary Construction

**Algorithm 1:** Construction of Noun-Term Vocabulary

**Input:** $P_{DT}(t_\tau) = \{p_\tau\}_{\tau \in N}$ – set of posts in the forum thread at time step $t_\tau$;

**Input:** $S_{DT} = \{w_k\}_{k \in N}$ – the subject of the forum thread;

**Output:** $V^d_{DT}(t_\tau)$ – noun-term vocabulary at time step $t_\tau$;

```
1:  Ω ← MiningDependencyRelations (P_DT(t_τ)) // Ω = {(h_m, d_n)}_{m,n∈N}
2:  for each word w_k in S_DT do
3:    if (checkNoun(w_k) and length(w_k) > 3) then
4:      w_k ← lemmatization(w_k)
5:      w_k ← lowercase(w_k)
6:    else
7:      S_DT ← S_DT \ w_k
8:    end if
9:  end for
10: for each pair (h_m, d_n) in Ω do
11:   sim ← Σ_{w_k∈S_DT} sim_wu(h_m, w_k)
12:   if sim != 0 and h_m ∉ V^d_DT(t_τ) then
13:     V^d_DT(t_τ) ← V^d_DT(t_τ) ∪ (h_m)
14:   end if
15: end for
```

The extraction of pairs (noun_term, opinion_word) by using the dependency relations considered for opinion mining

The identification of the nouns terms in the subject of the forum thread

We remove the noun terms that are not relevant to the subject of the forum thread

# Proposed System

- Step 4: User-user similarity matrix construction

| Opinion Lexicon | Opinion Lexicon Dimension | Sentiment Categories | Characteristics |
|---|---|---|---|
| **SentiWordNet 3.0 (based on WordNet 3.0)** | 155,287 words from 117,659 synsets | Each word has a score ranging between 0 and 1 for each positive, negative, or neutral sentiment category. | • It is semi-automatically generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **Micro-WNOp (based on WordNet 2.0)** | 1,960 words from 1,105 synsets | Each word has a score ranging between 0 and 1 for each positive, negative, or neutral sentiment category. | • It is manually generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **MPQA Subjectivity Lexicon** | 8,221 words | Each word is in one of the following sentiment categories: positive, negative, both (positive and negative), or neutral. | • It is semi-automatically generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **Bing Liu's Opinion Lexicon (based on WordNet 2.0)** | 6,786 words | Each word is in one of the following sentiment categories: positive or negative. | • It is semi-automatically generated.<br>• It does not distinguish between parts of speech of the opinion words. |

# Proposed System

- Step 4: User-user similarity matrix construction

| Opinion Lexicon | Opinion Lexicon Dimension | Sentiment Categories | Characteristics |
|---|---|---|---|
| **SentiWordNet 3.0 (based on WordNet 3.0)** | 155,287 words from 117,659 synsets | Each word has a score ranging between 0 and 1 for each positive, negative, or neutral sentiment category. | • It is semi-automatically generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **Micro-WNOp (based on WordNet 2.0)** | 1,960 words from 1,105 synsets | Each word has a score ranging between 0 and 1 for each positive, negative, or neutral sentiment category. | • It is manually generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **MPQA Subjectivity Lexicon** | 8,221 words | Each word is in one of the following sentiment categories: positive, negative, both (positive and negative), or neutral. | • It is semi-automatically generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **Bing Liu's Opinion Lexicon (based on WordNet 2.0)** | 6,786 words | Each word is in one of the following sentiment categories: positive or negative. | • It is semi-automatically generated.<br>• It does not distinguish between parts of speech of the opinion words. |

# Proposed System

- Step 4: User-user similarity matrix construction

| Opinion Lexicon | Opinion Lexicon Dimension | Sentiment Categories | Characteristics |
|---|---|---|---|
| **SentiWordNet 3.0 (based on WordNet 3.0)** | 155,287 words from 117,659 synsets | Each word has a score ranging between 0 and 1 for each positive, negative, or neutral sentiment category. | • It is semi-automatically generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **Micro-WNOp (based on WordNet 2.0)** | 1,960 words from 1,105 synsets | Each word has a score ranging between 0 and 1 for each positive, negative, or neutral sentiment category. | • It is manually generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **MPQA Subjectivity Lexicon** | 8,221 words | Each word is in one of the following sentiment categories: positive, negative, both (positive and negative), or neutral. | • It is semi-automatically generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **Bing Liu's Opinion Lexicon (based on WordNet 2.0)** | 6,786 words | Each word is in one of the following sentiment categories: positive or negative. | • It is semi-automatically generated.<br>• It does not distinguish between parts of speech of the opinion words. |

# Proposed System

- Step 4: User-user similarity matrix construction

| Opinion Lexicon | Opinion Lexicon Dimension | Sentiment Categories | Characteristics |
|---|---|---|---|
| **SentiWordNet 3.0 (based on WordNet 3.0)** | 155,287 words from 117,659 synsets | Each word has a score ranging between 0 and 1 for each positive, negative, or neutral sentiment category. | • It is semi-automatically generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **Micro-WNOp (based on WordNet 2.0)** | 1,960 words from 1,105 synsets | Each word has a score ranging between 0 and 1 for each positive, negative, or neutral sentiment category. | • It is manually generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **MPQA Subjectivity Lexicon** | 8,221 words | Each word is in one of the following sentiment categories: positive, negative, both (positive and negative), or neutral. | • It is semi-automatically generated.<br>• It distinguishes between parts of speech of the opinion words. |
| **Bing Liu's Opinion Lexicon (based on WordNet 2.0)** | 6,786 words | Each word is in one of the following sentiment categories: positive or negative. | • It is semi-automatically generated.<br>• It does not distinguish between parts of speech of the opinion words. |

# Proposed System

- Step 4: User-user similarity matrix construction

**Algorithm 3:** Opinion Word Sentiment Mining using SentiWordNet / Micro-WNOp

**Input:** w – opinion word;
**Input:** pos – part of speech for opinion word *w*;
**Output:** a value ∈ {-1, 0, +1} – indicates the sentiment of opinion word *w*, where: -1 denotes a negative sentiment, 0 denotes a neutral sentiment, and +1 denotes a positive sentiment;

```
1:  if !FindOpinionWord(w, pos) then
2:    return 0
3:  end if
4:  S ← GetSenses(w, pos)
5:  for each s ∈ S do
6:    difference ← GetPositiveScore(w, s) - GetNegativeScore(w, s)
7:    if difference = 0 and GetPositiveScore(w, s) != 0 then
8:      continue
9:    end if
10:   if difference = 0 and GetPositiveScore(w, s) = 0 then
11:     return 0
12:   end if
13:   if difference > 0 then
14:     return +1
15:   end if
16:   if difference < 0 then
17:     return -1
18:   end if
19: end for
20: return +1
```

**Algorithm 4:** Opinion Word Sentiment Mining using Bing Liu's opinion lexicon

**Input:** w – opinion word;
**Output:** a value ∈ {-1, +1} – indicates the sentiment of opinion word *w*, where: -1 denotes a negative sentiment and +1 denotes a positive sentiment;

```
1:  if !FindOpinionWord(w) then
2:    return 0
3:  else
4:    if w ∈ GetPostiveList() then
5:      return +1
6:    end if
7:    if w ∈ GetNegativeList() then
8:      return -1
9:    end if
10: end if
```

**Algorithm 5:** Opinion Word Sentiment Mining using the MPQA subjectivity lexicon

**Input:** w – opinion word;
**Input:** pos – part of speech for opinion word *w*;
**Output:** a value ∈ {-1, 0, +1} – indicates the sentiment of opinion word *w*, where: -1 denotes a negative sentiment, 0 denotes a neutral sentiment, and +1 denotes a positive sentiment;

```
1:  if !FindOpinionWord(w, pos) then
2:    return 0
3:  else
4:    polarity ← GetPolarity(w)
5:    if polarity = "positive" then
6:      return +1
7:    end if
8:    if polarity = "negative" then
9:      return -1
10:   end if
11:   if polarity = "neutral" then
12:     return 0
13:   end if
14:   if polarity = "both" then
15:     return +1
16:   end if
17: end if
```

# Dataset

- We use the Internet Argument Corpus (IAC)
  - http://www.4forums.com
  - A dataset freely available
  - Each discussion thread is saved in the JSON format

**Table 1.** Statistics on the experimental corpus.

| Forum Thread Identifier | Forum Thread Subject | Time Steps ($t_j$) | Number of Users ($U_{DT}(t_j)$) |
|---|---|---|---|
| Forum Thread 1 | "proof of God's existence" | $t_{49}$ | 7 |
| | | $t_{100}$ | 14 |
| | | $t_{149}$ | 15 |
| Forum Thread 2 | "Atheists - America's greatest threat?" | $t_{60}$ | 18 |
| | | $t_{121}$ | 19 |
| | | $t_{192}$ | 20 |

# Results for Forum Thread 1

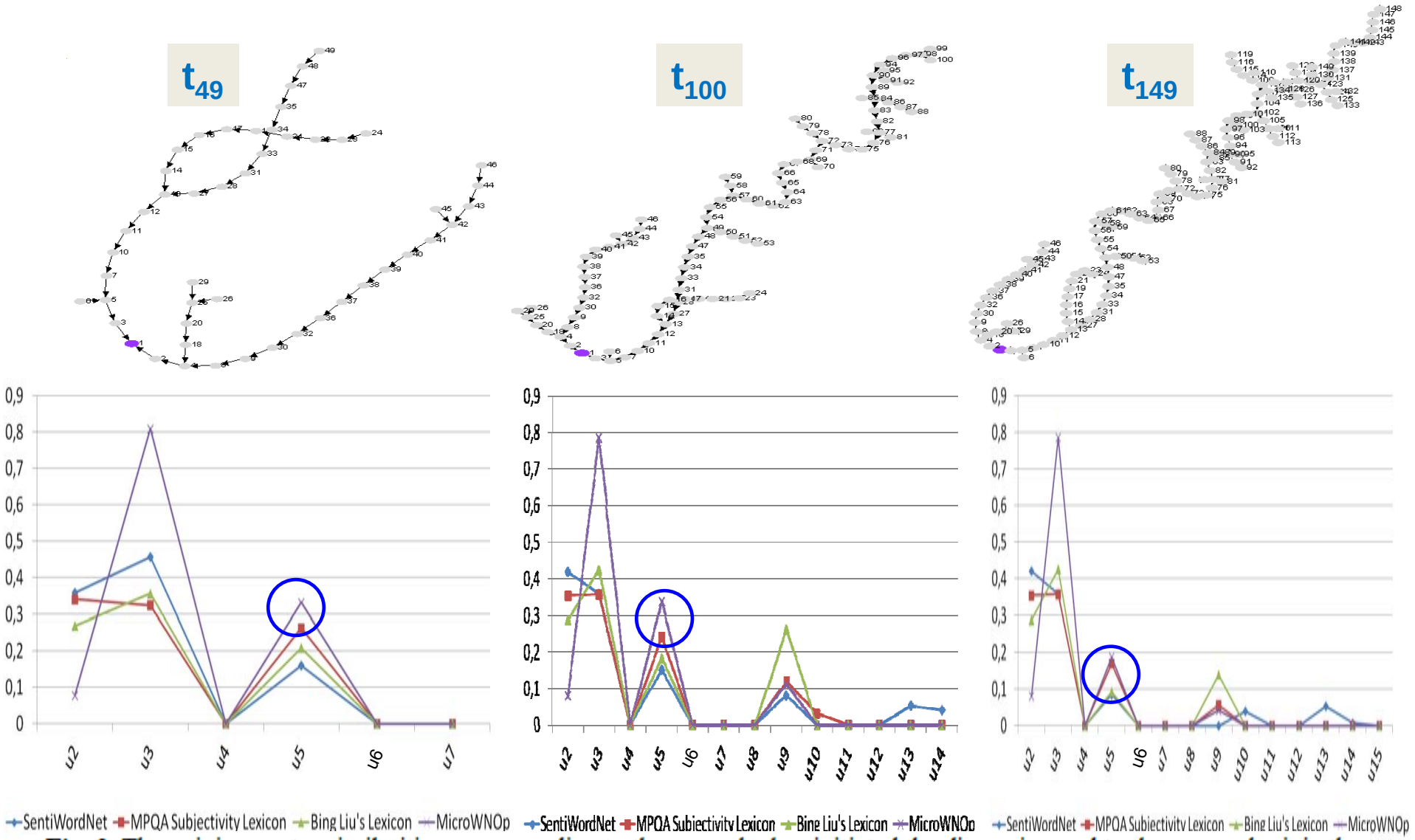



**Fig. 3.** The opinion vector similarities corresponding to the user who has initiated the discussion and to the users who join the discussion over time

# Results for Forum Thread 2

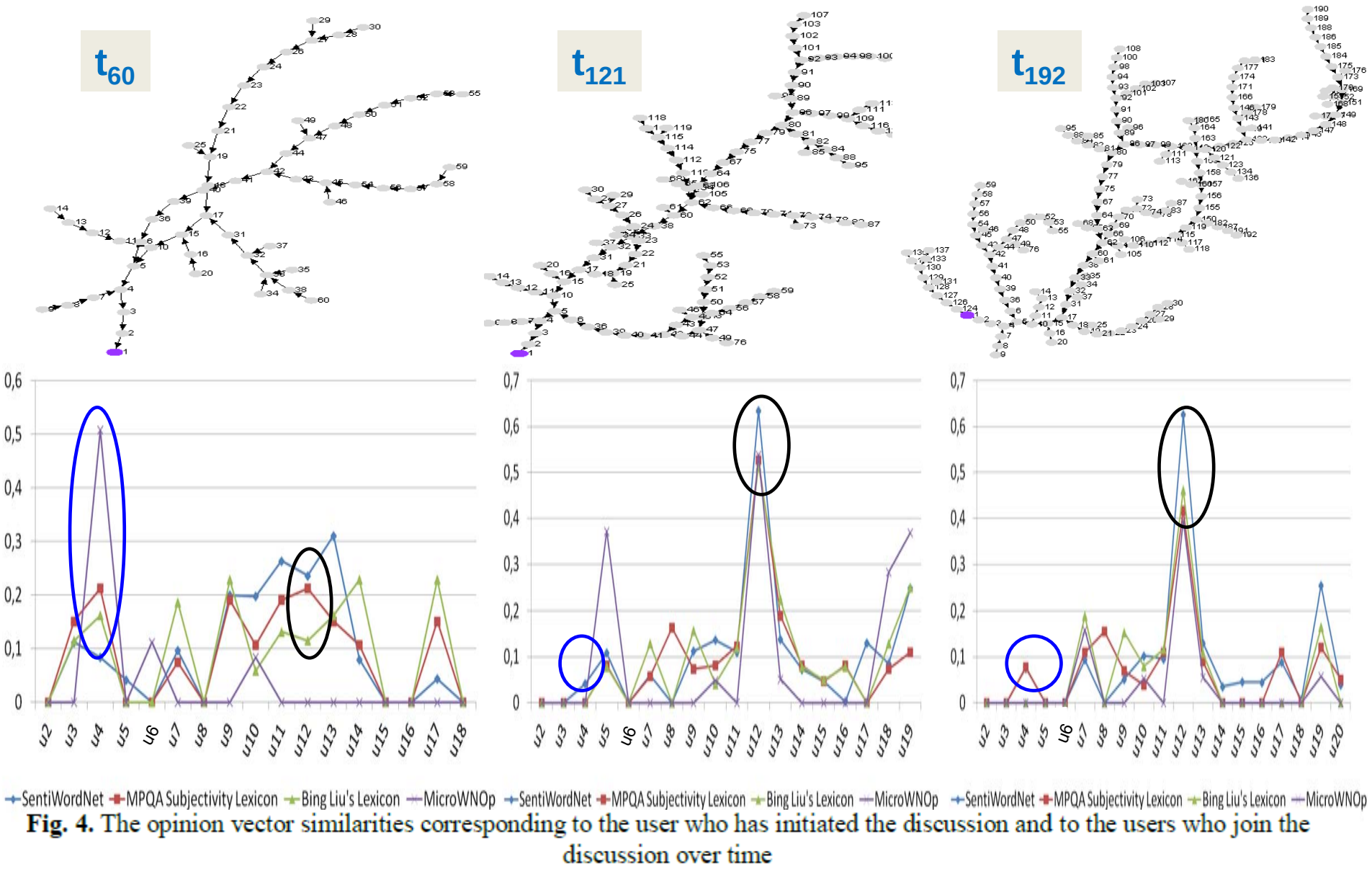



**Fig. 4.** The opinion vector similarities corresponding to the user who has initiated the discussion and to the users who join the discussion over time

# Results for Forum Thread 1

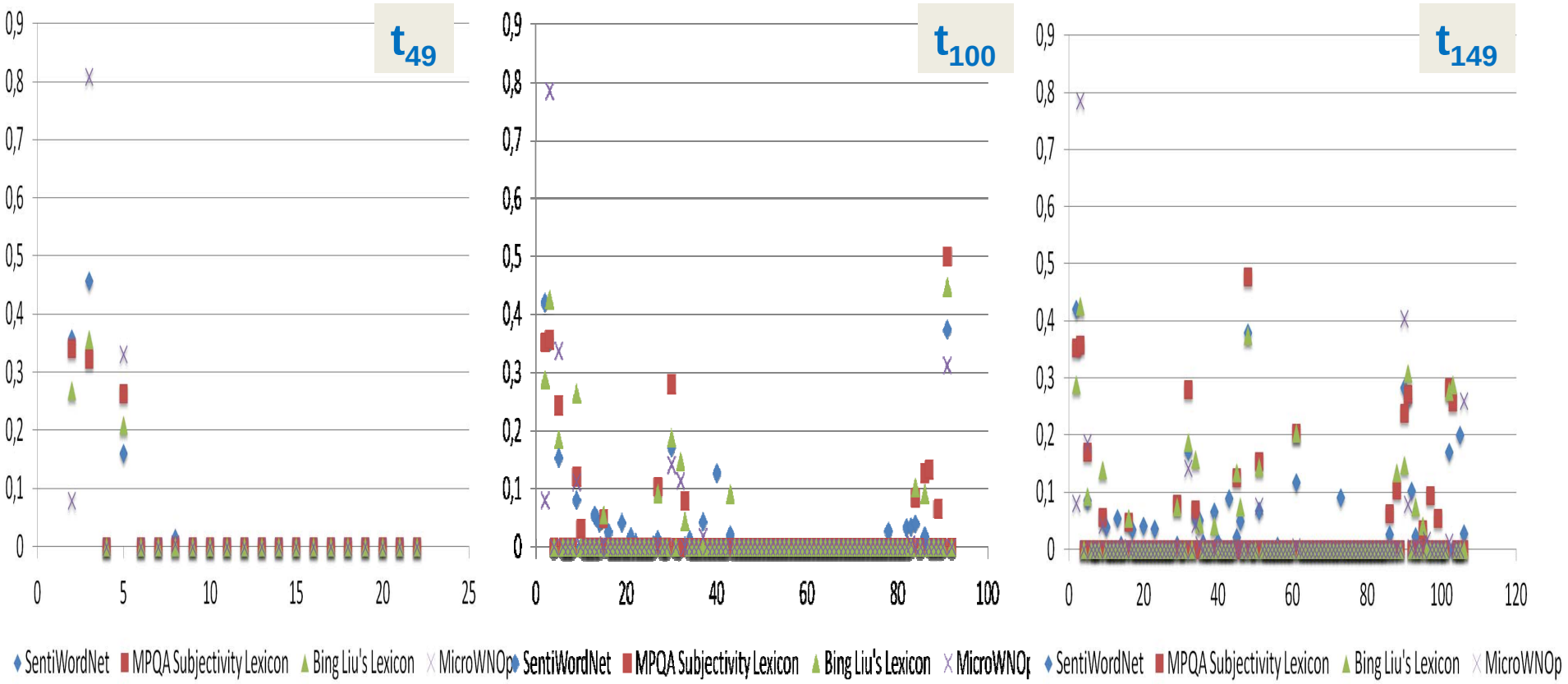



**Fig. 5.** The values of the user-user similarity matrix $B_{U\text{-}U}$

# Results for Forum Thread 2

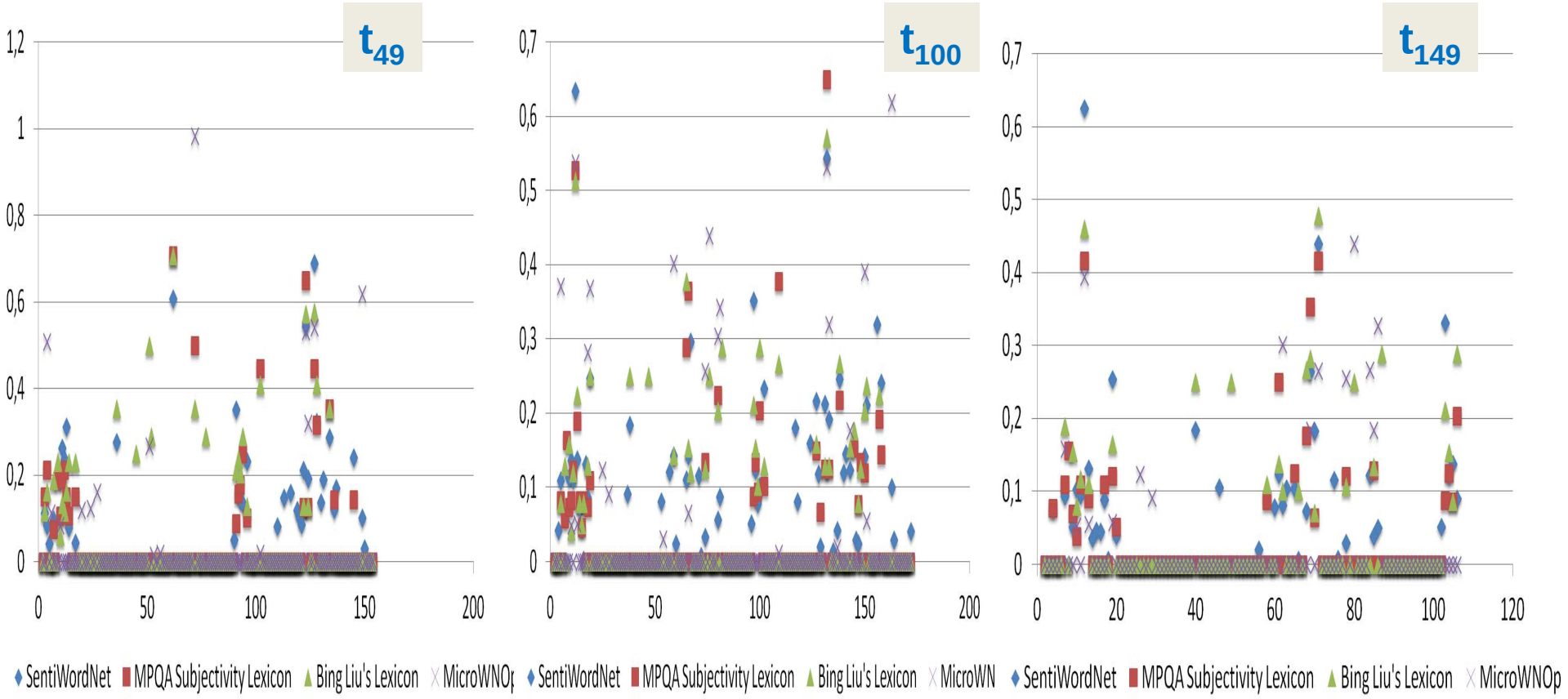



**Fig. 6.** The values of the user-user similarity matrix $B_{U\text{-}U}$

# Conclusions

- More research on information propagation, but little on opinion propagation in social media
- It is useful to solve the opinion propagation problem because of its multiple applications in the real world
- We proposed a user-level opinion propagation analysis method in online forum threads, by combining opinion mining and natural language processing techniques
- The results of our method for the opinion propagation problem depend on the opinion mining techniques we use

# Future Work

- Application of our opinion propagation analysis method to other online forum threads
- Analysis of users' behavior in the opinion propagation process in online forum threads
- Study of the opinion propagation problem in different types of social media, such as blogs and online social networks (e.g. Twitter), in order to detect and describe the opinion propagation process in these media

# Thank You!

**Questions**