# Projecting of the text document searching system: indexation using graph structures.

Paweł Szołtysek

22 September 2008

# Agenda



Agenda
Introduction
Conceptual Graphs
Our approach
Abilities
Conclusion

# Introduction to the topic
# Case study: HowTo startup



- ✔ Current information searching methods;
- ✔ Role of wikipedia in current world;
- ✔ One website to rule them all.

# Introduction to the topic
# Case study: HowTo startup



- ✔ Current information searching methods;
- ✔ Role of wikipedia in current world;
- ✔ One website to rule them all.

# Introduction to the topic
# Case study: HowTo startup



- ✔ Current information searching methods;
- ✔ Role of wikipedia in current world;
- ✔ One website to rule them all.

# Information access
# Case study: HowTo startup



- ✔ Navigation searching (categories, links);
- ✔ Attribute searching (tags, date created, author);
- ✔ Full text searching.

# Information access
# Case study: HowTo startup



- ✔ Navigation searching (categories, links);
- ✔ Attribute searching (tags, date created, author);
- ✔ Full text searching.

# Information access
# Case study: HowTo startup



- ✔ Navigation searching (categories, links);
- ✔ Attribute searching (tags, date created, author);
- ✔ Full text searching.

# Full text searching
# Case study: HowTo startup



## Is semantics a solution?

- ✔ Might be relevant...
- ✔ Might be fast...
- ✔ ...might take much resources...

# Full text searching
# Case study: HowTo startup



Is semantics a solution?

- ✔ Might be relevant...
- ✔ Might be fast...
- ✔ ...might take much resources...

# Full text searching
# Case study: HowTo startup



Is semantics a solution?

- ✔ Might be relevant...
- ✔ Might be fast...
- ✔ ...might take much resources...

# Full text searching
# Case study: HowTo startup



Is semantics a solution?

- ✔ Might be relevant...
- ✔ Might be fast...
- ✔ ...might take much resources...

# Full text searching
# Case study: HowTo startup



Is semantics a solution?

- ✔ Might be relevant...
- ✔ Might be fast...
- ✔ ...might take much resources...

Let's try graphs!

# What about conceptual graphs?



Conceptual graphs are well described, both for saving knownledge and searching systems.

There are many disadvantages of auto-creating CG's from pure text:

- ✔ Takes much resources.
- ✔ Disambiguations can occur.
- ✔ Best current available ideas bases on other semantic solutions.

# What about conceptual graphs?



Conceptual graphs are well described, both for saving knownledge and searching systems.

There are many disadvantages of auto-creating CG's from pure text:

- ✔ Takes much resources.
- ✔ Disambiguations can occur.
- ✔ Best current available ideas bases on other semantic solutions.

# Our approach to the searching.



Users will sacrifice accuracy for speed.

# Our approach to the searching.



Users will sacrifice accuracy for speed.
We want to simplify the idea of conceptual graphs.

# Our approach to the searching.



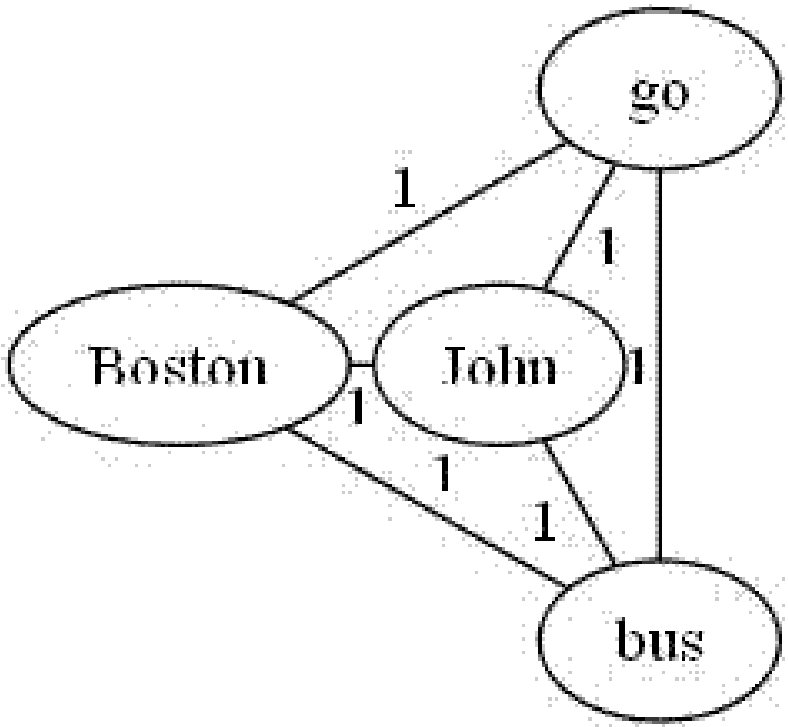
Users will sacrifice accuracy for speed.

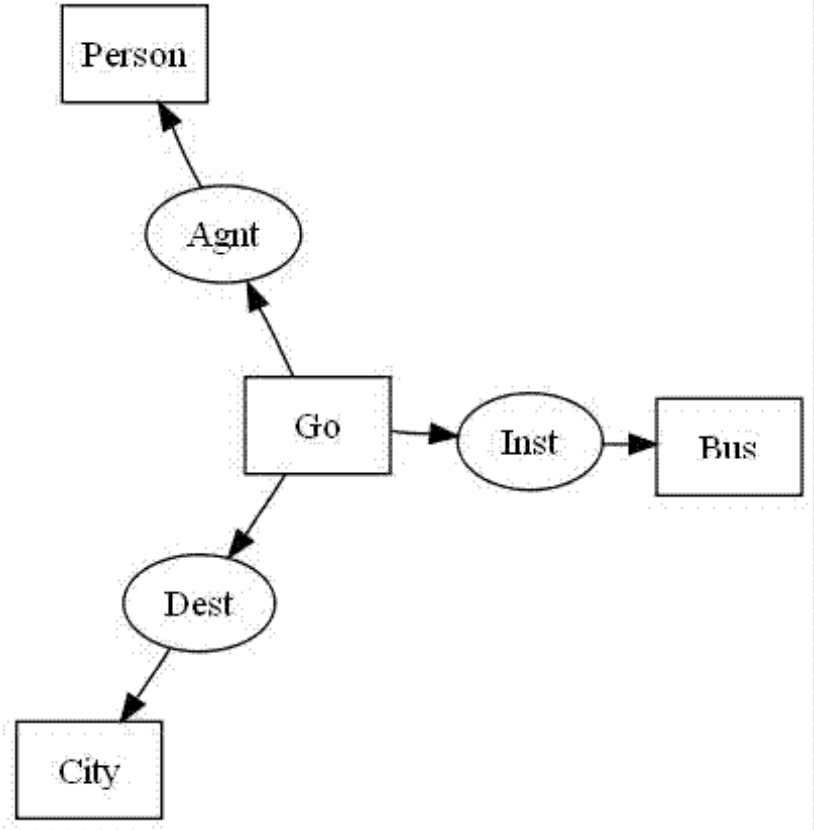
We would like to have vertices as terms, connected themselves with edges labelled by value of frequency and neighbourhood. To make searching easier, those weights will be normalized and quantified.

# Conceptual graphs compared.



John is going to Boston by bus.

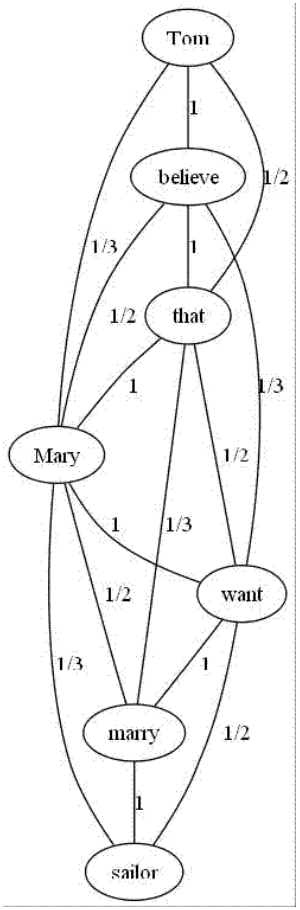


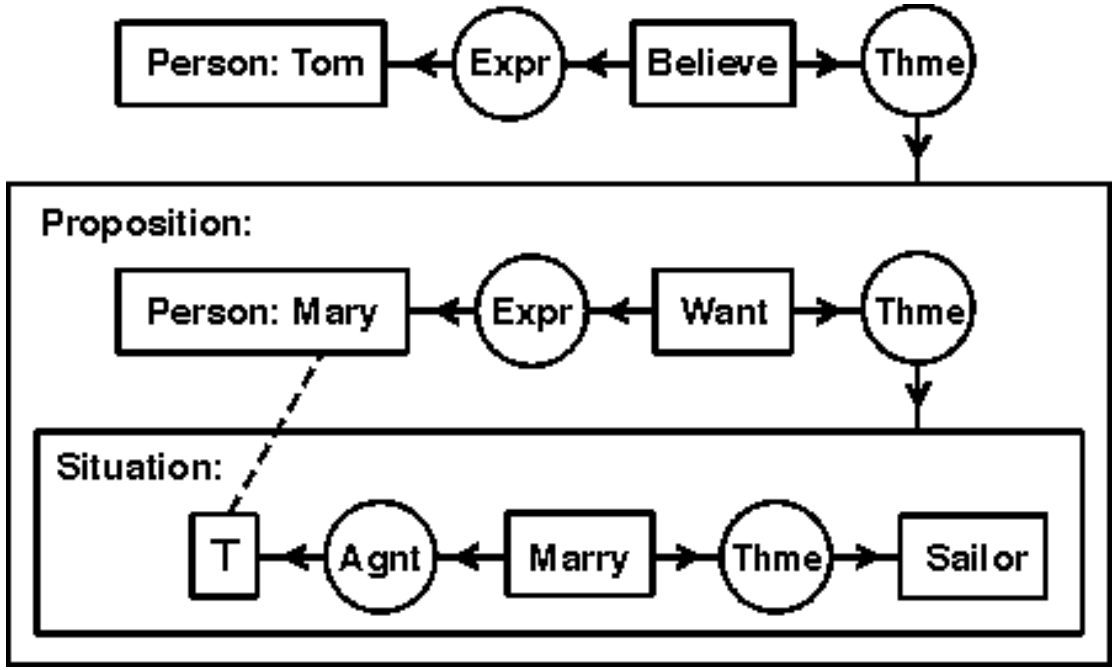



# Conceptual graphs compared.



Tom believes that Mary wants to marry a sailor.



# Core indexation functions



$\delta : D \rightarrow S'$ - document $\rightarrow$ subset of set of words which are in language with repetitions.

$\phi : S \rightarrow T'$ - basic form of each word in document is being found.

$\gamma : T' \rightarrow G$ - basing on terms our graph structure is being created.

# Core indexation functions



$\delta : D \rightarrow S'$ - document $\rightarrow$ subset of set of words which are in language with repetitions.

$\phi : S \rightarrow T'$ - basic form of each word in document is being found.

$\gamma : T' \rightarrow G$ - basing on terms our graph structure is being created.

# Core indexation functions



$\delta : D \rightarrow S'$ - document $\rightarrow$ subset of set of words which are in language with repetitions.

$\phi : S \rightarrow T'$ - basic form of each word in document is being found.

$\gamma : T' \rightarrow G$ - basing on terms our graph structure is being created.

# Core indexation functions



$\delta : D \rightarrow S'$ - document $\rightarrow$ subset of set of words which are in language with repetitions.
$\phi : S \rightarrow T'$ - basic form of each word in document is being found.
$\gamma : T' \rightarrow G$ - basing on terms our graph structure is being created.

Search engine bases on user's query, and all queries (such as boolean queries) can be adopted for proposed system.

# Abilities of the system: Recommendations



Recommendation system based on graph comparing.

Two types of measures: similarity and inclusion.

# Abilities of the system: Recommendations



Recommendation system based on graph comparing.

Two types of measures: similarity and inclusion.

# Similarity measure



Combination of conceptual and relational similarities - comparing vertices and edges.

$$\theta_s(G_1, G_2) = \frac{1}{2}\left(\frac{2n(G_i)}{n(G_1) + n(G_2)} + \frac{2m(G_i)}{m_{G_i}(G_1) + m_{G_i}(G_2)}\right) \quad (1)$$

Continuous, defined on range $[0, 1]$, with value 1 when graphs are conceptually identical, and 0 when they are completely different.

# Inclusion measure



Modification of similarity measure.

$$\theta_c(G_1, G_2) = floor(\frac{n(G_i)}{n(G_1)} + \frac{m(G_i)}{m_{G_i}(G_1)}) - floor(\frac{n(G_i)}{n(G_2)} + \frac{m(G_i)}{m_{G_i}(G_2)}) \quad (2)$$

Digital, with results within $-1, 0, 1$. If intersection is identical to $G_1$ it's $1$, $G_2$ - $-1$, and in all other cases $0$.

# Conclusion



- ✔ Presented indexation method bases on different types of basic index.
- ✔ First implementation tests confirmed usefulness of the system.
- ✔ Developed recommendation system has various ways of usage.

Further work

- ✔ Recomendation system as proposed after small changes can be useful while searching for plagiarisms.
- ✔ Size of neighbourhood and weights of edges could depend on the document (it's size, diversity).