Enhanced Information Access to Social Streams through Word Clouds with Entity Grouping

Martin Leginus\textsuperscript{1}, Leon Derczynski\textsuperscript{2} and Peter Dolog\textsuperscript{1}

\textsuperscript{1}Department of Computer Science, Aalborg University
Selma Lagerlofs Vej 300, 9200 Aalborg, Denmark

\textsuperscript{2}Department of Computer Science, University of Sheffield
S1 4DP, United Kingdom

20.05.2015
<table>
<thead>
<tr>
<th>Agenda</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word clouds for social streams</td>
</tr>
<tr>
<td>Entity redundancies and motivation</td>
</tr>
<tr>
<td>Grouping entities</td>
</tr>
<tr>
<td>Graph based word cloud generation</td>
</tr>
<tr>
<td>Synthetic evaluation</td>
</tr>
<tr>
<td>User study</td>
</tr>
<tr>
<td>Discussions, Conclusions and Future work</td>
</tr>
</tbody>
</table>
Word clouds generated from social streams

- A visual retrieval interface depicting the most important terms of a dataset.
- Word clouds provide means to minimize an information overload when browsing social media.

Figure: The user interface of FeedWinnower system.

Figure: Tweetmotif user interface for exploratory search.

Word clouds generated from social streams

Figure: Eddi - summarization interface of user timeline tweets.

Motivation

Word clouds generated from terms often not meaningful - enrich with named entities. (clouds with entities perceived as more useful (Finn et.al 2010))

The football team Manchester United can be referred to as MUFC, Man U, Red Devils or the Reds.

Redundancies decrease a quality of word clouds (decreased diversity, user confusion and limited browsing experience).
Goals

1. Condense divergent terms referring to the same entity.
2. Maximize a diversity of word clouds to provide a broad overview of topics.

Research hypothesis: Word clouds with grouped named entities improve coverage, relevance and diversity.
The process of word clouds generation

1. Data collection
2. Data preprocessing
3. Word cloud generation
Grouping named entities

1. Recognise named entities (NER) and disambiguate them (entity linking) using the TextRazor service.

- **TextRazor.**

<table>
<thead>
<tr>
<th>Entity</th>
<th>Confidence Score</th>
<th>Relevance Score</th>
<th>DBpedia Type</th>
<th>Freebase Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paul Scholes /m(0x0446d)</td>
<td>10.9901</td>
<td>0.297968</td>
<td>Agent Person Athlete SoccerPlayer</td>
<td>/people/person /soccer/football_player /sports/pro_athlete /people/measured_person</td>
</tr>
<tr>
<td>Manchester United F.C. /m(0x056ff)</td>
<td>29.9917</td>
<td>0.0996789</td>
<td>Agent Organisation SportsTeam SoccerClub</td>
<td>/business/sponsored_recipient /soccer/football_team /organization/organization /award/award_nominee /business/employer /sports/sports_team /award/award_winner /sports/sports_award_winner /sports/professional_sports_team</td>
</tr>
</tbody>
</table>

2. Find alternative names for the recognised entity.
3. Perform lemmatisation:
4. Using the aliases, build a term cluster for each entity.
5. Find canonical names for entities.
Grouping named entities

1. Recognise named entities (NER) and disambiguate them (entity linking) using the TextRazor service

2. Find alternative names for the recognised entity.
(We employ a Freebase KB)
Grouping named entities

1. Recognise named entities (NER) and disambiguate them (entity linking) using the TextRazor service.
2. Find alternative names for the recognised entity.
3. Perform lemmatisation: Group together all the inflicted forms of a word to exploit only the base form of the term e.g., fan, fans.
4. Using the aliases, build a term cluster for each entity.
5. Find canonical names for entities.
Grouping named entities

1. Recognise named entities (NER) and disambiguate them (entity linking) using the TextRazor service.
2. Find alternative names for the recognised entity.
3. Perform lemmatisation:
4. Using the aliases, build a term cluster for each entity e.g., mufc, manchester united, man united, red devils, devils.
5. Find canonical names for entities.
Grouping named entities

1. Recognise named entities (NER) and disambiguate them (entity linking) using the TextRazor service
2. Find alternative names for the recognised entity.
3. Perform lemmatisation:
4. Using the aliases, build a term cluster for each entity
5. Find canonical names for entities. Represent the cluster *mufc*, *manchester united*, *man united*, *red devils*, *devils* with Manchester United F.C.
Grouping named entities

1. Recognise named entities (NER) and disambiguate them (entity linking) using the TextRazor service
2. Find alternative names for the recognised entity.
3. Perform lemmatisation:
4. Using the aliases, build a term cluster for each entity
5. Find canonical names for entities.

Proceed with word cloud generation.
Graph-based word cloud generation

Terms extracted from tweets used for a graph creation. When two terms (vertices) co-occur at least $\alpha$ times, two directed edges are introduced $t_1 \rightarrow t_2, t_2 \rightarrow t_1$

Tweets are short, hence a parameter $\alpha$ is set to 0.
Graph-based ranking

Stochastic traversal of terms graph estimates an importance of a term $t$. Iterative stationary probability is defined as:

$$\pi(v)^{(i+1)} = (1 - \beta) \left( \sum_{u=1}^{d_{\text{in}}(v)} p(v|u)\pi^{(i)}(u) \right) + \beta \vec{p}_p$$

(1)

User preferences can be encoded into a vector of prior probabilities $\vec{p}_p$. The resulting global rank of a term $t$ after convergence is:

$$I(t) = \pi(t)$$

(2)

Top-k ranked terms are then used for word cloud generation.
Divrank ranking

Intention is to increase the diversity of ranking
Transition probabilities change over time - "rich gets richer principle"

Divrank ranking

A transition probability from a node $u$ to node $v$ at time $T$ is:

$$ p_T(v|u) = (1 - \beta) \left( \frac{p_0(v|u) \cdot p_T(v)}{\sum_{v \in V} p(v|u) \cdot p_T(v)} \right) + \beta p_p $$

Divrank algorithm is useful for incorrectly disambiguated entities e.g., BBC, BBC NEWS, BBC NEWS WORLD.
Evaluation metrics

A term \( t \) links to tweets \( Tw_t \)

\( Tw_{tq} \) is the set of all tweets that are associated with a query phrase \( t_q \)

Coverage indicates how many tweets are retrievable from the given word cloud.

\[
\text{Coverage}(WC_k) = \frac{\big| \bigcup_{t \in WC_k} Tw_t \big|}{|Tw_{tq}|},
\]  

Overlap captures the extent of redundancies i.e., how many terms link to the same tweet.

\[
\text{Overlap}(WC_k) = \text{avg}_{t_i \neq t_j} \frac{|Tw_{t_i} \cap Tw_{t_j}|}{\min\{|Tw_{t_i}|, |Tw_{t_j}|\}},
\]
Evaluation metrics I

Mean Average Precision:

1. A word cloud is transformed into a query $Q_{WC_k}$.
2. Retrieve and rank tweets matching the query.
3. Measure Mean Average Precision (MAP).

Ranking function is Okapi BM25:

$$S(tw, Q_{WC_k}) = \sum_{q_i \in Q_{WC_k} \cap tw} c(q_i, Q_{WC_k}) \cdot TF(q_i, tw) \cdot IDF(q_i)$$  \hspace{1cm} (5)

The function $c(q_i, Q_{WC_k})$ returns a weight of the term $q_i$.

Components $TF(q_i, tw)$ and $IDF(q_i)$ are calculated in the standard way.
TREC2011 microblogging collection with relevance judgements. Tweets rated as relevant or highly relevant are considered equally relevant.

**PgRankTerms** (baseline) estimates a global importance of terms (extracted from tweets and transformed into a graph).

**MFE:** selects top-\(k\) most popular recognized entities.

**MFEA:** selects top-\(k\) most popular recognized entities grouped with their aliases.

**PgRankTermsEntities:** ranks top-\(k\) phrases from the graph which contains extracted terms and grouped recognized entities.
Results

Figure: Word clouds with grouped entities attain higher Coverage and MAP.
Figure: Divrank decreases redundancies and also improve Coverage wrt. to the baseline.
User study

- Are word clouds with named entities perceived as more relevant and diverse by the users?
- Do measured synthetic metrics correlate with the ratings of relevance and diversity by users?
160 distinct relevance ratings, 89 positive towards word clouds with named entities, 27 neutral ratings and 44 for the baseline generated word clouds. Diversity ratings, 73 positive towards word clouds with named entities, 51 neutral ratings and 36 for the baseline generated word clouds.
MAP metric predicts extrinsic human evaluations of cloud quality.
False positives from NER affect relevance ratings e.g., a cloud for “Super Bowl, seats" contained “Super (2010 American film)".

Imprecise named entity disambiguation (e.g., BBC, BBC News, BBC News World) increases redundancies.

Due to subjective nature of the crowdsourcing task, we disregarded a user qualifying phase.
Conclusions

- A technique that groups aliases of the same entity and represents them with a canonical term.
- Significantly decreased redundancy and significantly higher coverage than the baseline.
- User study supports that word clouds with grouped named entities are significantly more relevant and diverse than baseline.
- MAP metric predicts extrinsic human evaluations of cloud quality.