Exploring Significant Interactions in Live News

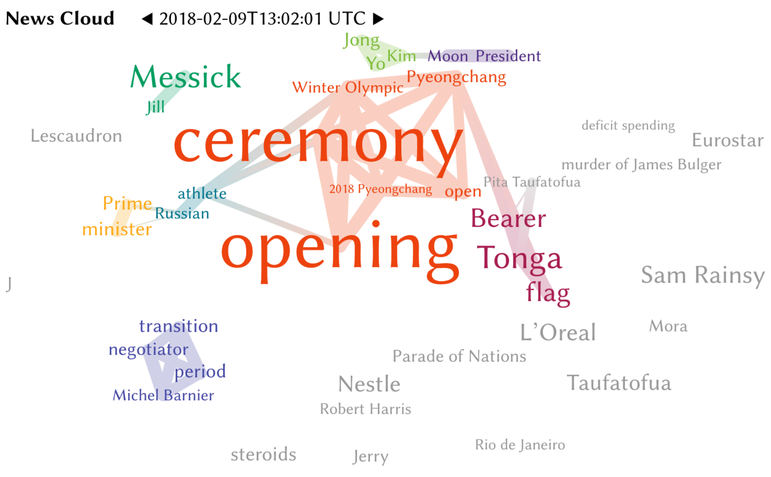
Erich Schubert **Andreas Spitz** **Michael Gertz**

Database Systems Research Group, Heidelberg University, Germany
{schubert,spitz,gertz}@informatik.uni-heidelberg.de

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

Abstract

News monitoring is of interest to detect current news and track developing stories, but also to explore what is being talked about. In this article, we present an approach to monitoring live feeds of news articles and detecting significant (co-)occurrences of terms compared to a learning background corpus. We visualize the result as a graph-structured semantic word cloud that uses a stochastic neighbor embedding (SNE) based layout and visualizes edges between related terms. We give visual examples of our prototype that processes news as they are crawled from dozens of news sites.



Motivation

Interactions are more important than single terms:

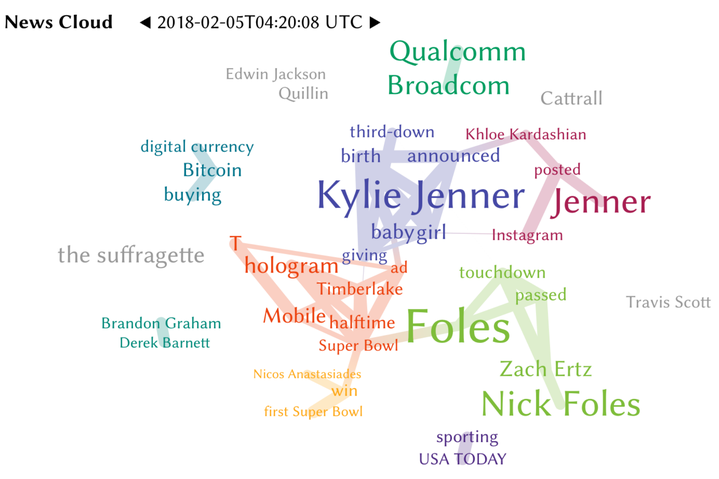
- "Edward Snowden" and "NSA"
- "Great Britain" and "Russia"
- "Sergej Skripal" and "Nowitschok"

Objective

Our objectives are to:

- analyze a **live stream of news articles**
- detect **significant interactions** of entities and topics
- visualize the result for **explorative** data analysis

Example



- edges indicate significiant cooccurrences
- size visualizes current significance
- colors denote clusters

Method

Text Processing

We extract entities, verbs, and nouns (using CoreNLP). Other words are not considered for cooccurrences.
Entities are detected and linked using a model trained on Wikipedia, which merges synonyms and abbreviations, such as "EU" and "European Union" into a single token with the canonical name.
We use a Gaussian weighting scheme for cooccurrences with weight $w_d = \exp(-d^2/2\sigma^2)$ for words at a token distance of d, and use $\sigma = 4$ with a window width of 12.

Learning the Background Model

To estimate expected (co-)occurrence frequencies, we use a hashing-based approximation (MinHash / Bloom filter based, as used by SigniTrend), with the initial values obtained from Wikipedia.
We then continually incorporate the documents we have already seen into our background model to adapt it to current news. We update our background model at the end of each micro-batch such that it combines a fraction $99\% = 1 - \eta$ of the previous model, and a fraction $\eta = 1\%$ based on the documents we just processed.
Using the hash table, this approach can be implemented efficiently and does not require storing or revisiting documents outside the current micro-batch.

Judging Significance

The basic idea of significance is to compare the observed (co-)occurrence weight $c(t)$ with the expected frequency $E[t]$, but we have to account for a number of biases.
In particular, for a term never seen before, we have $E[t] = 0$ and then $c(t)/E[t] = \infty$. Therefore, we need to introduce pseudocounts (Laplace correction) to the equation. Because we evaluate this for every token (co-)occurrence, we also need a prior that models randomly choosing k terms.

We define the relevance ratio $r(t)$ and probability $p(t)$:

$$\text{relevance ratio: } r(t) = \max\left\{0, \frac{c(t) - \beta_D}{E[t] + \beta_C}\right\} \cdot p'$$

$$\text{relevance probability: } p(t) = \frac{r(t)}{r(t)+1}$$

where:

$c(t)$ = observed weighted (co-)occurrence
$E[t] = \min_i \text{table}[\text{hash}_i(t)]$ expected (co-)occurrence
$\beta_D = \frac{1}{2}/|D|$ batch size bias (pseudocounts)
$\beta_C = \frac{1}{2}/|C|$ corpus size bias (pseudocounts)
$p' = k/|W|$ prior prob. to choose k words out of $|W|$

The expected frequency $E[t]$ using the hash functions provides a guaranteed lower bound of the true frequency, with a low collission probability for frequent terms (but overestimation on rare terms such as spelling errors).

Clustering and Visualization

Words are clustered with average-linkage hierarchical clustering, using $p(t)$ as similarity. Clusters of at least two tokens are extracted. Unclustered tokens are colored gray.

Based on the cooccurrence significance matrix using the $p(t)$ above, we use Stochastic Neighbor Embedding (SNE) to compute "anchor" positions for each word.
In the browser, we use D3.js to optimize the final layout using a force directed graph where words are attracted to their "anchor" position, but repulse each other to avoid word overlap.

Try it yourself!

newsir-demo.ifi.uni-heidelberg.de
Use the arrow buttons to navigate.
Click the date to jump to a date of interest.

Changes from the paper version to the live demo:

- More sources monitored
- Boilerplate removal is slightly improved
- We no longer split tokens at a hyphen (e.g., J-20 plane on 2018-02-09, T-Mobile on 2018-02-05)
- Entites with a hyphen now correctly linked (e.g., Moon Jae-in on 2018-02-09)
- We fall back to CoreNLP NEC to get more entites, even if we cannot link them to Wikidata

Performance

Stream volume:

- 5,900 crawled pages / day in February
- 2,900 usable English articles / day in Feburary
- 1,700 usable German artciles / day in Feburary
- 650,000 English articles analyzed
- 440,000 German articles analyzed

Analysis speed:

- Intel Core i5-4570 (2013), single-core:
- English: 3.52 articles / second (304,000 / day)
- German: 1.74 articles / second (150,000 / day)
- Performance difference due to CoreNLP

Future Work

Visualization improvements:

- Support drill-down into clusters in the UI
- Display example sentences and snippets
- Improve layout: also use non-significant cooccurrences
- Persistent clusters over time
- Storyline generation
- User study to evaluate benefits

Analysis:

- Self-learning boilerplate removal
- Near-duplicate detection
- Duplicate paragraph removal
- Learn emerging entities and names
- Streaming graph analysis of cooccurrences
- Emerging clusters and cliques

References:
E. Schubert, A. Spitz, M. Weiler, J. Geiß, and M. Gertz. "Semantic Word Clouds with Background Corpus Normalization and t-distributed Stochastic Neighbor Embedding". In: *CoRR* abs/1708.03569 (2017)
A. Spitz and M. Gertz. "Terms over LOAD: Leveraging Named Entities for Cross-Document Extraction and Summarization of Events". In: *ACM SIGIR*. 2016
E. Schubert, M. Weiler, and H.-P. Kriegel. "SigniTrend: Scalable Detection of Emerging Topics in Textual Streams by Hashed Significance Thresholds". In: *ACM KDD*. 2014
C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky. "The Stanford CoreNLP Natural Language Processing Toolkit". In: *ACL System Demonstrations*. 2014

NewsIR'18 Workshop at the 2018 European Conference on Information Retrieval (ECIR 2018) — March 25–29 2018, Grenoble, France