Inverse Document Density: A Smooth Measure for Location-Dependent Term Irregularities

Dennis THOM   *Harald BOSCH*   *Thomas ERTL*
Institute for Visualization and Interactive Systems, University of Stuttgart, Germany
`Dennis.Thom@vis.uni-stuttgart.de, Harald.Bosch@vis.uni-stuttgart.de,`
`Thomas.Ertl@vis.uni-stuttgart.de`

ABSTRACT
The advent and recent popularity of location-enabled social media services like Twitter and Foursquare has brought a dataset of immense value to researchers in several domains ranging from theory validation in computational sociology, over market analysis, to situation awareness in disaster management. Many of these applications, however, require evaluating the a priori relevance of trends, topics and terms in given regions of interest. Inspired by the well-known notion of the *tf-idf* weight combined with kernel density methods we present a smooth measure that utilizes large corpora of social media data to facilitate scalable, real-time and highly interactive analysis of geolocated text. We describe the implementation specifics of our measure, which are grounded in aggregation and preprocessing strategies, and we demonstrate its practical usefulness with two case studies within a sophisticated visual analysis system.

TITLE AND ABSTRACT IN ANOTHER LANGUAGE, L_2 (OPTIONAL, AND ON SAME PAGE)

Inverse Dokumentdichte: Ein glattes Maß für ortsbezogene Termnutzungsunregelmäßigkeiten

Das Aufkommen und die derzeitige Beliebtheit von ortsbezogenen Diensten der Sozialen Medien wie Twitter und Foursquare haben einen Datensatz von immensem Wert für Forscher verschiedener Domänen, von der Aussagenvalidierung in der Soziologie, über Marktanalysen, bis zur Situationseinschätzung im Katastrophenschutz, geschaffen. Viele dieser Anwendungen erfordern jedoch eine Einschätzung der a priori Relevanz von Trends, Themen und Termen für gegebene geographische Regionen. Basierend auf der Idee hinter dem bekannten Tf-idf-Maß, präsentieren wir eine geglättetes Maß, welches, durch die Ausnutzung großer Korpora bestehend aus Daten der Sozialen Medien, die skalierbare, echtzeitfähige und voll interaktive Analyse von geokodierten Texten ermöglicht. Wir beschreiben die Details der Umsetzung unseres Maßes, welche auf Aggregations- und Vorverarbeitungsstrategien beruht, und wir zeigen seine praktische Anwendbarkeit durch zwei Fallbeispiele mit Hilfe eines elaborierten Systems zur visuellen Analyse.

KEYWORDS: tf-idf, term density, geolocated corpora, Visual Analytics, Twitter, social media.

KEYWORDS IN L_2: TD-IDF, Termdichte, geokodierte Korpora, Visual Analytics, Twitter, Soziale Medien.

Proceedings of COLING 2012: Technical Papers, pages 2603–2618,
COLING 2012, Mumbai, December 2012.

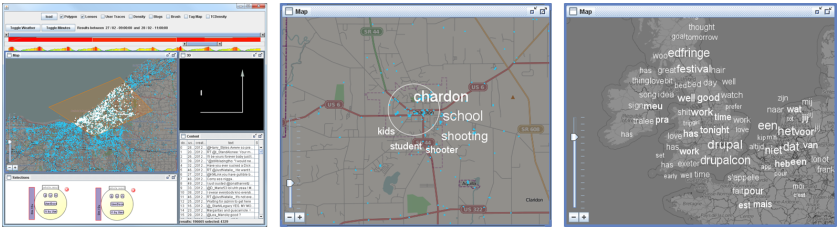


Figure 1: Left: The ScatterBlogs system for visual social media analysis. Interactive exploration techniques are used to examine aggregated message contents. Middle: *content lens* — a circle which can be interactively moved over the map to show the most frequent terms within a region and timeframe. Right: Cluster analysis is used to detect spatiotemporal clusters of similar topic usage to display them on a map.

1 Introduction

Every time Twitter users write a message on their GPS-enabled mobile device, they can attach precise location information. Each day more than 3 million documents are produced this way. Within one year, people from all over the world have generated a corpus comprising more than 1.3 billion[1] geolocated messages. Such location-enriched text data has tremendous value for researchers and analysts in several fields ranging from theory validation in computer sociology to location aware market analysis. Most notably, this data source has opened important application domains for research in situation awareness and disaster management, since the community of social media users can serve as a global ‘sensor network’ of potential incident reporters of critical events (Sakaki et al., 2010).

Due to the high complexity and volume of data, automated means for language processing and information mining are often complemented by highly interactive visualization tools (MacEachren et al., 2011; Marcus et al., 2011) that help analysts to explore and filter relevant messages and detect important localized events like river floods, wildfires, typhoons, hurricanes, infectious diseases or shooting incidents (Starbird and Palen, 2010; Mendoza et al., 2010; Sakaki et al., 2010; Hughes and Palen, 2009; Chew and Eysenbach, 2010; Palen et al., 2009; Heverin and Zach, 2010). Several of these tools have shown that aggregated representations of keywords and topics of large message volumes from selected or automatically determined spatiotemporal ranges can be of great value for precise situation assessment and large scale anomaly indication. For example, the ScatterBlogs Visual Analytics system (Bosch et al., 2011; Thom et al., 2012), depicted in Figure 1, provides means to aggregate large volumes of textual data in selected areas and indicates spatiotemporal topic clusters with geolocated Term Clouds.

Relevant keywords in such systems are often detected through their high term-occurrence counts compared to the counts of other terms in the vicinity. It has been a challenge, however, to evaluate whether documents or even individual terms are actually anomalous outliers within the specified area and timespan or just perennially prominent terms like stopwords or certain keywords that are frequently used in the examined region or throughout the whole Twitter

[1] Tweets can either have a user defined geotag (e.g. `London`) or precise latitude/longitude coordinates. In this work we only consider the latter ones, comprising a subset of 0.7 billion messages.

network. As a result, relevant and anomalous documents and keywords are often obscured by falsely highlighted keywords resulting from regular day-to-day chatter. Conventional solutions for this problem rely on the use of *tf-idf* and similar measures, which indicate whether a keyword is specifically important within a selected document set or whether it is also frequent within the corpus of all documents. However, when looking at geolocated documents the situation is quite different. Besides globally prominent terms, which can easily be detected by *tf-idf*, there are also many terms that are only prominent within a certain region. Thus, when examining a certain geospatial area and timeframe of documents, the importance of certain prominent and unusual terms can still be easily obscured by numerous and ordinary terms for the area.

To address this problem we present a smooth and scalable technique for geospatial term relevance normalization. Based on the combination of *tf-idf* with kernel density methods, a complete one-year corpus of geolocated Twitter messages is evaluated to determine the a priori probability that a given term is contained in a document composed at a given location. The intuition behind our measure is quite simple: In the *idf* part of the *tf-idf* measure, the number of documents in which a term appears is counted in order to put the term frequency in relation to the sum of documents in the corpus. In contrast, our measure sums for a given location the derived probabilities that a document containing the term could have appeared at this point and puts it in relation to the sum of derived probabilities that any document could have appeared at this point. The outcome of this is the (im)probability that a term is contained in a message appearing at any given point, allowing to assess the abnormality of observed term occurrences in examined document sets.

The details and a formal definition of this concept will be given in Section 3. An important contribution of our paper is a scalable implementation to calculate, store and quickly retrieve the normalization values based on adaptive grid aggregation techniques, described in Section 4. Since the proposed technique was particularly developed for large scale interactive document exploration, the performance details are discussed in Section 5 and its practical applicability is evaluated in Section 6. We conclude with a discussion of the results, final remarks and an outlook on future work.

2 Related Work

Work related to the described approach can be found mainly in two areas, which will be addressed in this section. First, geolocated resources can be used to create meta-documents for specific locations in order to geographically tag new resources according to their similarity to these meta-documents. Second, the geographic information of resources can be exploited to establish a ‘geo-ranking’ of search results lists.

Several approaches propose a prediction of a resource’s geolocation inferred from the textual content or associated tags by features such as toponyms, geographic and cultural features, and stylistic and dialectic differences. Wing & Baldridge (2011) discretize the world with a regular grid of equal degree and calculate the term probabilities per document, grid cell and corpus using different geolocated document collections, i.e. Wikipedia articles and Twitter messages. Given a new and unlocated document, they can calculate the similarity between the document term-distribution and the cells’ term-probabilities and choose the closest cell as the probable document location. Roller et al. (2012) follow the same approach but construct an adaptive grid from a space-partitioning k-d-tree trained on the same document collections for efficiency and higher resolutions in densely populated regions. For each grid cell, a pseudo-document is assembled from all training documents that lie within the cell and again their similarities to documents of unknown origin are calculated. This can lead to a decreasing quality of the measure when the resolution is increasing, due to an overfitting of the data when the pseudo-documents will become small and specific for the high resolution cells. In contrast, our approach uses a smoothing kernel, such that the measure's quality is increasing with higher resolutions. Serdyukov et al. (2009) present a similar approach related to the tags associated with user uploaded photos. They consider a bag-of-tags model for each grid cell and combine it with a smoothing strategy and additional codified knowledge about geolocations from services such as GeoNames[2]. However, all these approaches examine the situation where an unknown document is given and the most probable location of its origin has to be found. Therefore it is reasonable to estimate the probability that a certain term or document *appears* at a certain location. In contrast, our approach estimates the probability that a document written at a certain location *contains* a certain term. What seems to be just a slight difference, leads to a completely new application domain, as will be demonstrated in our case studies.

Zhang et al. (2010) also present a geospatial extension of the *tf-idf* measure in the context of tag centric query processing for geolocated web 2.0 resources. The approach tries to identify regions where each of the query tags is covered by at least one nearby web 2.0 resource. The purpose of the measure is to rank regions within the result set according to how characteristic the tags are for each region. Due to the focus on querying, their use case and therefore the measure itself and its implementation differs from ours in the following aspects. The geospatial extension can be seen as replacing documents by regions and the corpus by the set of all regions. A term is therefore characteristic for a region when it is *frequently* observed in the region but *infrequently* used globally. As our work focuses on exploring large data sets with potentially huge amounts of frequent but irrelevant chatter, we need a measure for normalization with respect to long-term keyword densities. Our approach contrasts term densities with document densities for each individual point of the world. Therefore, a term can still achieve a high score for a specific region even if it is a globally common word. Furthermore, the approach of Zhan et al. utilizes a fixed grid in order to approximate a smooth measure and a tagged resource can only influence directly neighboring cells by a two-step degradation function. This limits the spatial scalability and allows only rough resolutions when dealing with global data.

The World Explorer visualization tool (Ahern et al., 2007) identifies representative labels for geolocated photo collections. The available photos are geospatially clustered, independent of their associated tags, to form potential map label locations for each zoom level and map tile individually. Afterwards, a representative tag is selected for each cluster by evaluating an adapted *tf-idf* measure. Similarly to Zhang et al., the tag frequency within a cluster is related to its overall frequency within one tile. With this approach, tags like *San Francisco* are relevant on the scale of the state of California, but insignificant on a detailed tile of San Francisco itself. Eisenstein et al. (2010) present a latent geographic topic model to identify words with high regional affinity, geographically coherent linguistic regions, and regional variations of general topics. The model can be seen as a latent Dirichlet allocation with regional corruptions of the base topics. The model was applied to 380k twitter messages and achieved a good performance in locating users by their allocation to a regional topic.

3 Model

In information retrieval and language processing, the *tf-idf* measure is used to evaluate the relevance of a term for a given document within a given corpus (Jones, 1972; Manning et al.,

[2] `http://www.geonames.org`

2008). Given a document d and a term t, the number of occurrences $tc_{t,d}$ of t in d is determined and normalized by document length to compute the term frequency $tf_{t,d} = \frac{1}{|d|} tc_{t,d}$. This results in the relative prominence of t within d. A high term prominence, however, is no indicator that the term is also important to the document, since common terms like `the`, `she` or `like` are prominent within most documents. Therefore, one also has to compute the inverse document frequency based on the corpus D of all documents, from which d was taken:

$$idf_{t,D} = \log \frac{|D|}{|\{d | d \in D \wedge t \in d\}|} \tag{1}$$

The *idf* can be seen as an indicator for the a priori probability that t appears in documents drawn from D - the higher the probability, the lower the value. The *tf-idf* is computed by simply multiplying the values:

$$tfidf_{t,d,D} = tf_{t,d} * idf_{t,D} \tag{2}$$

Moving from regular document sets to the domain of geolocated messages, we find a very diverse corpus with contents ranging over several different regional topics, characteristics and languages. For example, in most major cities around the globe the city's own name as well as the names of individual districts are usually mentioned in hundreds of Twitter messages every day. The same is true for regional points of interest or words from languages and dialects which are only used in distinct parts of the world. In order to estimate, whether a term is actually important or anomalous for a particular map area and time span, it is not sufficient to compare its local term frequency against the global *idf* value. Globally, the term `Denver` could have a similar or even lower *idf* value as `shooter`. What is needed instead is a measure that compares the term count of the examined message set with the estimated inverse document frequency of all messages that have been written in the appropriate region and in a larger timeframe than that of the message set.

To estimate this inverse document density for arbitrary map locations from the sparsely distributed message corpus, we utilize kernel based density estimation (KDE), which approximates probability distributions from discrete point data, and integrate it with the traditional *tf-idf* measure. Characteristics of the KDE and the formal definition of the new measure will be detailed in the following two subsections.

3.1 Kernel Density Estimation

It is a well-known problem to derive a continuous probability density f from a finite sample data set $X = \{x_1 \dots x_n\}$. For example, X could be a list of locations of crime reports for a major city. In KDE techniques (Rosenblatt, 1956; Parzen, 1962), the so-called density estimator for the data set is constructed as follows:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{d(x, x_i)}{h}\right) \tag{3}$$

In this equation, $d(x, y)$ is a distance metric, e.g. the Euclidean distance in case of samples from $\mathbb{R}^n$. The function K is the kernel and it is used together with the bandwidth h to assign a weighted value to each x_i depending on its distance from x. For the sake of simplicity one can also write the equation using subscript notation $K_h(u) = \frac{1}{h} K(\frac{u}{h})$. Selected kernel functions usually have their maximum at $u = 0$, are rapidly decreasing with higher u, and integrate to 1. A Gaussian kernel is used in our implementation but other choices are also conceivable:

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{u^2}{2}\right) \tag{4}$$

The purpose of using these functions is to reflect the probability that a given sample could deviate a given distance u from its actual location. For example, if a crime was committed somewhere along a street at point x, without prior knowledge, it could as well have happened 5, 10 or 100 meters farther away with decreasing probabilities. By summing and normalizing all these spatially decreasing probabilities from all samples, the constructed function $\hat{f}(x)$ gives us a relative estimate of the probability that a crime can happen at an arbitrary location.

3.2 Measure Definition

Similar to the crime example above, the KDE principle can be applied to a sample of term occurrences from Twitter messages m_t containing term t at location x to inspire the estimation of term occurrence densities at arbitrary map locations. In the presented measure, KDE is employed to define the localized version of the inverse document frequency *idf* as basis for the localized *tf-idf* measure. It is assumed, that a given corpus G of geolocated messages has been collected over a large temporal range, e.g., one year, to stabilize the corpus against seasonal characteristics. We first define a measure for local term density and normalize it by the corresponding local document density to calculate the local inverse document density:

For a given term t let $G_t = \{m \in G : t \in m\}$ be the subset of messages from G that contain t and let $loc(m) \in \mathbb{R}$ be the location of message m in the coordinate space. Furthermore, let K_h be a kernel with fixed bandwidth h. For a given location $x \in \mathbb{R}$ we call

$$td_t(x) = \sum_{m \in G_t} K_h(d(x, loc(m))) \tag{5}$$

the term density of t at x. Note that these are absolute and not relative densities as they are not normalized with $\frac{1}{|G_t|}$. In order to allow a cross-comparison of different terms, the terms' densities are normalized at each location using the term independent document density at the same location:

$$dd(x) = \sum_{m \in G} K_h(d(x, loc(m))) \tag{6}$$

Finally, and analogous to the *idf*, we call

$$idd_t(x) = \log \frac{dd(x)}{td_t(x)} \tag{7}$$

the inverse document density of t at x.

In these equations the distance function must be matched to the coordinate space that has been chosen to represent message locations. For the example in Section 3.1, we assumed a uniform grid coordinate system and thus the Euclidean metric was an appropriate choice. However, since Twitter messages are usually given in graticule coordinates (latitude, longitude), one can convert them to a uniform grid or use the haversine formula (Sinnott, 1984) to approximate the distance.

For the term frequency value *tf* there is a natural analogue in most application cases. However, compared to the *idd* these values cannot be precomputed, since we want to apply the measure to new messages as they arrive from a continuous real-time data stream. Furthermore, in the case of Twitter messages, which are bound to 140 characters, there will rarely be more than one occurrence of a specific term and it is thus not meaningful to calculate the term frequency for a single message. However, if a set of messages M is examined - e.g. within a user selected region and timeframe - one can build a localized *tf-idf* value for any term t by calculating the sum

$$\sum_{m \in M} tc_{t,m} * idd_t(loc(m)) \tag{8}$$

This equation properly reflects the relation of current prominence of the term versus its commonness at the message locations. For the sake of simplicity and computational cost, the value can be approximated by calculating the term frequency for a pseudo-document, generated by concatenating all documents in M, and multiplying it with the $idd_t(mean)$ at the centroid location $mean = \frac{1}{|M|} \sum_{m \in M} loc(m)$ of the documents.

4 Implementation

For the traditional *tf-idf* measure, it is expensive to compute the *idf*-part, as the whole corpus has to be analyzed. Therefore, the values are usually precomputed at once by iterating through the complete set of documents and terms within the corpus. Once this is done, one can quickly compute a *tf-idf*-vector for any given document by computing a term frequency value for each term $t \in d$ and multiplying it with its precomputed $idf_{t,D}$ value.

The computation of the $idd_t(x)$ values is even more expensive, because for any point x the sum of kernel-weighted distances between x and $loc(m)$ for all messages $m \in G$, and for the messages $m \in G_t$ respectively, has to be calculated. Furthermore, as we are looking at a continuous and thus infinite coordinate space, it is not feasible to precompute and store an $idd_t(x)$ value for every possible t and x.

For practical applications, however, there is no need to have an infinitely high spatial resolution of values. Therefore, a high resolution regular grid can be laid over the globe and the $td_t(x)$ and $dd_t(x)$ values can be compute at every cell center or vertex. Missing values between these points could then be calculated at runtime through interpolation. Nevertheless, to achieve cell-sizes below 0.5 kilometer at the equator line, a grid resolution of at least 80 000 × 40 000 cells would be needed for global coverage. If we assume that each value takes 4 Bytes of storage, this amounts to approximately 12 Gigabytes of data for every single term in the corpus. To counter these problems, we adhere to a grid construction strategy that is adaptive to regional requirements resulting from population density. This will be detailed in the next subsection. Section 4.2 explains how the *idd* values are actually computed for each generated grid cell using a technique called Splatting. Finally, we explain in Section 4.3 how the values are stored and quickly retrieved once they are needed by the application.

4.1 Grid Creation

In many regions of the world, like oceans, deserts or large rural areas, the occurrence of Twitter messages is sparse. At the same time a high Tweet frequency in major cities can be observed. Instead of using a regular grid with a fixed resolution, it is reasonable to use an adaptive grid with high resolution in densely populated areas and lower resolution elsewhere. This grid is

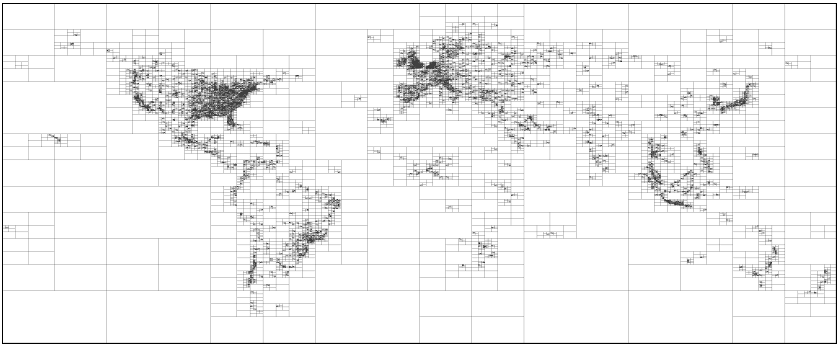


Figure 2: The result of adaptive grid creation, generated with a maximum depth of 16 and cell splitting as long as more than 50 messages were contained within a cell.

constructed by a recursive splitting mechanism generating a quadtree data structure (Finkel and Bentley, 1974). Since only Twitter users are of interest for this data set, the relevant ‘Twitter population’ density can be directly derived from message densities in the observed corpus. A data sample of 10 days taken uniformly distributed from one year of recorded Twitter messages was used for the construction of the grid. The initial, single cell comprises the complete coordinate space with $lat \in [-90, 90]$ and $lon \in [-180, 180]$. As long as more messages than a fixed threshold fall within one cell, we split it into four equally sized sub-cells and recursively apply the algorithm to each. A recursive path is terminated as soon as a predefined minimum cell-size of 0.5 kilometers is reached. The result of this method can be seen in Figure 2. For further computation, the complete recursive tree structure with the leaves representing the grid cells is stored. This way we can quickly ($O(\log |leaves|)$) find cells containing a given location x by recursively searching through the tree. Furthermore, a unique cell ID c_i is assigned to each cell and we use $loc(c_i)$ to denote the center-location of the cell.

4.2 Fast Value Computation

In a Gaussian kernel K_h, more than 99% of the area beneath the curve is within a $3h$-radius from the center. During the computation of the $td_t(x)$ and $dd(x)$ values, messages which are further away from x than this radius can be ignored, as they add little to nothing to the sum. Based on this observation, we chose a strategy called Splatting to realize a fast *idd* precomputation. The technique originates from volume rendering for 3D graphics (Westover, 1991). The concept can be considered as ‘throwing ink balls’ onto the grid at every message location, which results in a Gaussian footprint, a so-called *splat*. The local sums of the footprints at each grid-cell add up to the *td* and *dd* values. The concept is illustrated in Figure 3.

The algorithm works as follows. Let $grid = \left\{c_1, ..., c_i\right\}$ be the set of cell IDs of a grid data structure as created in Section 4.1. Furthermore, let $DD : grid \rightarrow \mathbb{R}$ and $TD_t : grid \rightarrow \mathbb{R}$ be initially empty Hash tables that map cell-ids to computed *dd* and td_t values. A given corpus G is processed according to the `splat` procedure of Listing 1. Thus, instead of iterating through the whole corpus for each grid-cell, to calculate the kernel weighted distances, we iterate through

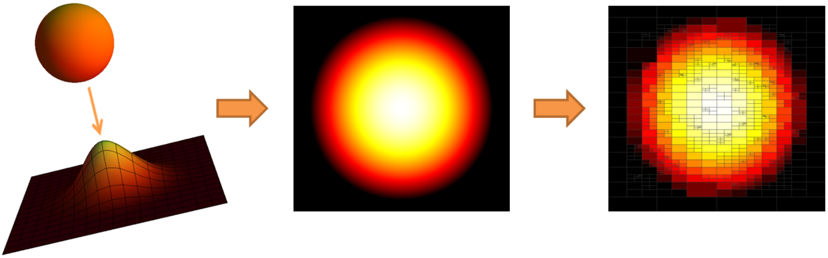


Figure 3: Left: Splatting process can be imagined as throwing ink balls onto the grid, resulting in a Gaussian signature. Middle, Right: The continuous splat is applied to the discrete grid. Each cell value is computed based on the distance from cell center to splat center.

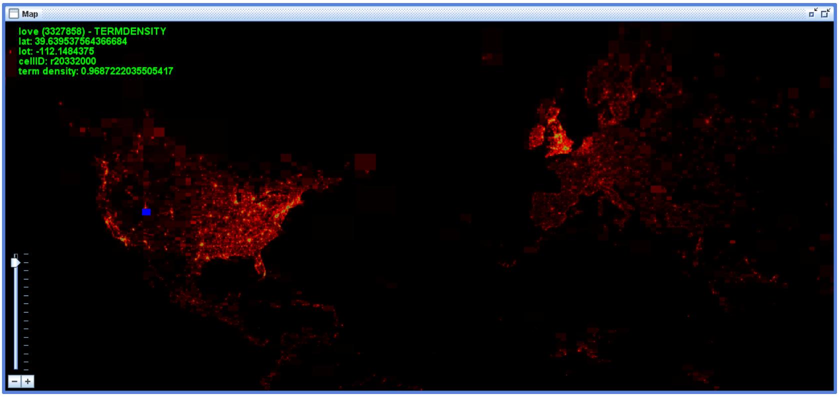
G just once and add for each $m \in G$ and $m \in G_t$ a Gaussian splat value to all affected hash table entries $DD(c)$ and $TD_t(c)$, where the center location $loc(c)$ is within a $3h$-distance from m.

```
procedure splat(G,TD,DD,grid) is
begin
  for each $m \in G$ do
    $impact\_area \leftarrow \{c \in grid : d(loc(c), loc(m)) \leq 3h\}$
    for each $c \in impact\_area$ do
      if $DD(c)$ =empty then
        $DD(c) \leftarrow K_h(d(loc(c), loc(m)))$
      else
        $DD(c) \leftarrow K_h(d(loc(c), loc(m))) + DD(c)$
      end if
    end for
    for each $t \in m$ do
      for each $c \in impact\_area$ do
        if $TD_t(c)$ =empty then
          $TD_t(c) \leftarrow K_h(d(loc(c), loc(m)))$
        else
          $TD_t(c) \leftarrow K_h(d(loc(c), loc(m))) + TD_t(c)$
        end if
      end for
    end for
  end for
end
```

Listing 1: Splatting Algorithm

As mentioned in Section 4.1 this ‘impact area’ can be found quickly using the quadtree data structure. Assuming a constant upper bound for the number of terms inside a Twitter message as well as the number of cells within a splat radius, the algorithm runtime can be estimated by $O(|G| * \log(|grid|))$. Also, in terms of memory management the hash tables provide an efficient means to store the data, as large volumes of grid cells in oceans and rural areas will be unaffected by the splats of most terms. For these areas, we avoid to redundantly storing the information $idd_t(c) = 0.0$.

To limit the duration of the precomputation phase (e.g. one day instead of three) it is reasonable



```
love (3327858) - TERMDENSITY
lat: 39.639537564366684
lot: -112.1484375
cellID: r20332000
term density: 0.9687222035505417
```

Figure 4: To evaluate the computed *TD* and *DD* results, we implemented a zoomable visualization, where individual values can be interactively examined by mouseover (blue rectangle). The result for the term density map of `love` can be seen in the window.

to restrict the computation to terms that have a certain minimum frequency. For example, in our case we included only terms that have at least 1000 mentions within a year — at application runtime, all other terms are then handled as if they occur for the first time with a default minimum term density of $td_t(x) = 1 * K_h(0)$. For evaluation purposes, the outcome of the splatting algorithm can be visualized by mapping the computed values to a color palette. The results for TD_{love} can be seen in Figure 4.

4.3 Fast Value Retrieval

The output of the splatting procedure will be a large set of filled hash tables TD_t for each term and *DD* for all documents, which can then be used for ad-hoc interpolation of the $idd_t(x)$ values at arbitrary points on the map. To actually apply the measure, two different modes were implemented. The first mode attaches the computed *idd* values directly to the twitter message as soon as it is stored. Currently, geolocated messages are collected from a continuous stream of approximately 3 – 4 million messages per day. Assuming that the number of terms per messages can be bounded by 20 terms, this means that the computation of one $idd_t(x)$ value has to be achieved in less than approx. 20 to 30 milliseconds. For future implementations the computation needs to be even faster, due to the still heavily increasing number of Twitter users.

The second mode is to retrieve the *idd* values only when they are needed inside an interactive application. In this case the retrieval from memory has to be fast enough to avoid interrupting fluid interactivity for the user. Thus it depends on the application what this means in terms of retrieval speed, i.e. for how many messages the value has to be computed in one interaction step. To actually store and fetch the values we used Apache Lucene[3], a widely known text search and storage system, and a custom built storage and query management. These methods will be reviewed in the next section.

[3] `http://lucene.apache.org/core/`

5 Implementation Performance

To evaluate its performance, the implementation was tested on a compute server driven by four Intel Xeon processors totaling to 40 physical cores, 128GB RAM, and SAS hard drives in a RAID 50 configuration. The *idd* values were precomputed based on a set of 732 895 428 geolocated Twitter messages collected between August 2011 and August 2012. The splatting algorithm described in Section 4.2 can easily be modified for parallel execution, such that it could fully benefit from the multicore capabilities of the test system. For a maximum depth of 18, the adaptive grid, described in Section 4.1, was created in less than 30 minutes and has about 300 000 cells. Based on this configuration, the complete precomputation process for the *TD* and *DD* tables was performed in less than 35 hours and it took approximately 200 Gigabytes to store the raw output.

A set of 1000 terms, drawn according to their overall frequency in the corpus, and 1000 randomly chosen cells of the grid were used to measure the retrieval speed of our storage solution. The adaptive grid tries to keep the amount of documents in each cell relatively constant; therefore the random cell node selection roughly reflects the document distribution. Because all terms share the same grid, only one instance of the quadtree needs to be held in memory for computing the cell ID for a given point x. The actual data is stored as mappings from cell IDs to term density values. We use Apache Lucene for storage and fast access to the precomputed $td_t(x)$ map. Each combination of term, cell ID, and *td* value was indexed by Lucene as a standalone *document*[4]. In order to access a value, the index for the term and cell ID combination is used to retrieve the document containing the value. This process takes 46ms on average (48ms for hits, 35ms for misses). Using parallelization with twenty threads, our implementation achieves an average of 4.8ms per value retrieval.

6 Case Studies and Evaluation

The practical usefulness of our measure was evaluated with scenario studies based on actual events and real world Twitter data. For this purpose the measure was integrated into existing tools for spatiotemporal text analysis and aggregation. In the following case studies we demonstrate how an analyst can employ and benefit from improved highlighting and term filtering based on the term normalization. Furthermore, we examine how our measure performs compared to traditional approaches.

6.1 Comic-Con 2012

The San Diego Comic-Con International is one of the largest annual conventions for comic books, science fiction/fantasy and other popular arts. From July 12 to July 15 2012 the main event was held at the San Diego Convention Center and several smaller events were co-located in nearby hotels and other venues. With more than 130 000 visitors swarming the area, observing and reporting show acts, autograph signings, meetings and other activity in the metropolitan area, the collected social media data is a perfect playground for large scale situation analysis.

For our case study we examined a set of 37 937 geolocated messages that were written in the San Diego area between 07/12 and 07/15 using the Visual Analytics system ScatterBlogs. Some of the system's core features integrate automated NLP methods for interactive visual analysis of aggregated text data. Here, we employed the *content lens*, a movable circle used to explore spatially plotted message contents by visualizing the most prominent terms. The

[4] Lucene stores documents as individually indexed string or numeric fields

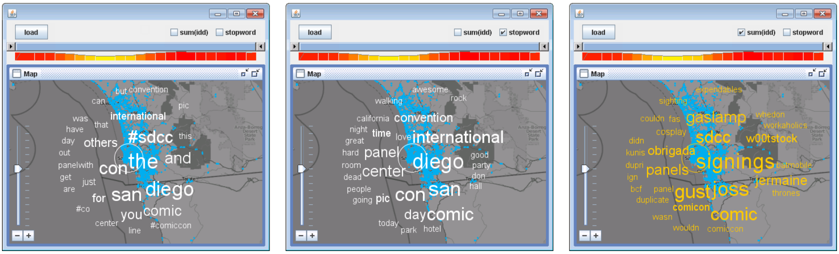

load
□ sum(idd) □ stopword
Map
but convention can pic was that international have day others #sdcc this out panelwith con the and get just are for san diego #co you comic center line #comiccon
load
□ sum(idd) ☑ stopword
Map
awesome walking rock california convention night time great love international hard panel good room diego party dead center don people san hall going pic con today day comic park hotel
load
☑ sum(idd) □ stopword
Map
expendables sighting whedon couldn fas gaslamp workaholics cosplay didn sdcc w00tstock kunis obrigada dupri panels signings batmobile ign jermaine bcf panel gustjoss duplicate thrones comicon wasn comic wouldn comiccon

Figure 5: In ScatterBlogs a circular lens can be dragged over the map to spatially explore aggregated message contents. Left: Terms from messages inside the lens are shown according to descending occurrences. Middle: An English stopword list is used to remove irrelevant terms. Right: The *idd*-sum is used to weight the terms.

result of applying this technique to the San Diego area can be seen in Figure 5. The screenshot on the left shows the lens operating in standard mode, i.e. all terms from messages within the selected spatial and temporal frame are extracted, counted, and then arranged with descending popularity. By fixing the lens to a specific location and selecting one of the labels, the corresponding messages are shown in a table and can be further investigated by the user. In most cases, however, the standard mode will not be very useful, as it is dominated by high frequency stopwords like `the` and `you`. The user can thus activate two filtered operating modes by selecting one of the checkboxes. In *stopword*-mode, as depicted on the image in the middle, a list of English stopwords is used to remove them from the aggregation. In this mode, at least some terms related to the event (`panel`, `international`, `comic`) achieve a high ranking and can be seen in the visualization. However, the top ranking terms are still dominated by terms of regional prominence (`san`, `diego`, `center`).

The image on the right shows the *content lens* with activated measure as described in Sections 3 and 4. Here one can see several terms indicating individual smaller sub-events and activities in connection with Comic-Con. Some selected examples are:

- `w00tstock` - The music and arts show `w00tstock` was co-located with the event.
- `signings` - Artists, authors and actors gave autographs in special signing sessions.
- `joss` - Joss Whedon (an American screenwriter) gave a press conference at that day.
- `batmobile` - Props from the batman movies were on display near the convention center.

Between the hundreds of high frequency terms, these terms only achieve such a high ranking because their high prominence for the spatiotemporal frame contrasts their average prominence within the region. Such sub-event indicators are often what an analyst is interested in, when examining localized large scale events and catastrophes.

6.2 Virginia Earthquake

In August 2011 the US state Virginia was hit by a magnitude 5.8 earthquake near Richmond. The first Tweets reporting the incident were written just seconds after the shockwave occurred.

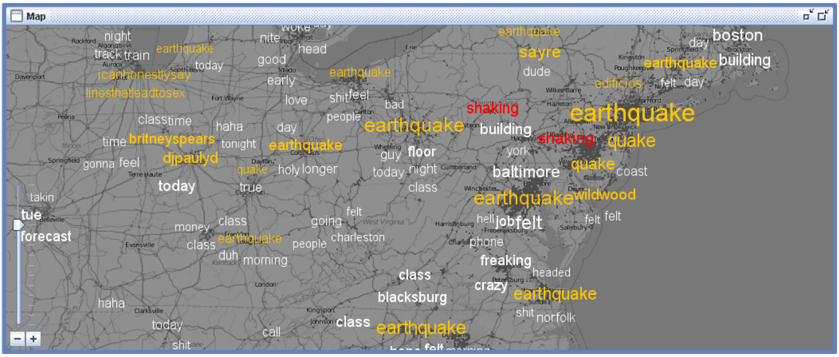


Figure 6: The *tag map* visualization: A k-means clustering scheme is employed to detect and display message clusters with similar timestamp, location and topic. Orange highlighting shows clusters/terms with high average *idd*.

To analyze this event, we use another tool of ScatterBlogs called *tag map*. It detects spatiotemporally dense clusters of messages with similar term usage and visualizes them by placing representative labels on the map. It follows the intuition that messages written by groups of local event observers will often result in this kind of *spatiotemporal content anomalies* (Thom et al., 2012) in the message streams. These anomalies can be used as an overview of the dataset and help to indicate suitable entry points for deeper investigation. Although the mechanism helps to detect many localized characteristics, it cannot distinguish whether a cluster results from a sudden and unexpected incident or from a regular venue - e.g. there could be a daily occurring cluster of messages containing `apple` at the location of an Apple store. Thus the overview showing several relevant terms is heavily cluttered with irrelevant terms that hinder the analysis.

To enhance the technique with our measure, a color coding was introduced that highlights topic clusters with high average *idd* values in orange. The threshold for highlighting is predefined based on the z-score of the *idd* value for all events. However, the analysts can also configure it to their needs using interactive controls. The result of applying the measure to a frame around the 23th of August can be seen in Figure 6. One can observe that relevant terms like `quake` and `shaking`, which are related to the sudden earthquake event, clearly stand out, while other prominent terms like `baltimore` and `blacksburg` are shown in the default color white. Besides the earthquake related terms one can also observe that `britneyspears` and `djpaulyd` are highlighted, because the artists happened to perform in Indianapolis during the examined timeframe.

6.3 Evaluation

We compared the results of our location aware measure against raw prominence and global *tf-idf* based term rankings for our observed scenario studies. For each scenario we used all messages within a predefined timeframe and area and computed the three measures for every term. For the 1000 most frequent terms a decision was made, whether they are specifically relevant for the examined scenario. Based on this labeling, we determined incremental precision/recall curves by ranking the terms according to the respective measure's values. An example for the Comic-Con case can be seen in Figure 7.

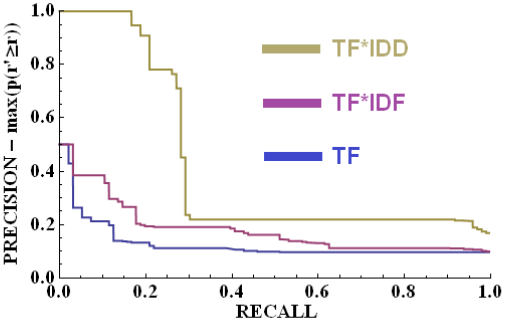


Figure 7: The graph shows interpolated precision ($p_{interp}(r) = \max_{r' \geq r} p(r')$) versus recall for Comic-Con 2012. From the messages written in the San Diego area the 1000 most prominent terms had to be ranked (96 were labeled as being relevant).

Our results show that our measure usually performs better in terms of precision for the top ranked terms and equally well or better for the lower ranked terms.[5] It is thus well suited for relevance based visualizations, where an analyst must quickly find interesting query-terms that could qualify for further investigation.

Conclusion and Perspectives

In this work a smooth measure for estimating term occurrence probabilities on a global scale was presented. Based on density adaptive large data aggregation, our measure can be employed in highly interactive environments and is suitable for real-time processing of streaming data. Because of the focus on term densities, our approach can be used in multilingual scenarios like the global Twitter corpus, as it is mostly language independent. It therefore eliminated the need for language specific stopword lists in the presented scenarios. Our case studies and evaluation have shown that the integration of the measure can lead to significant improvements for visual analysis systems.

Future work will encompass research in filtering techniques like loess smoothing to distinguish between seasonal, trending and unusual aspects of the data. In this context we will also examine means for explorative parameter steering to allow an interactive adaption of the measure's influence to tune results to the analysts' needs.

Acknowledgments

This work was funded by the German Federal Ministry for Education and Research (BMBF) and the European Commission as part of the VASA project and the FP7-project PESCaDO (FP7-248594). We would like to thank the reviewers for their valuable suggestions and comments.

[5] The relevance labeling and computed measure values for the scenario studies as well as further evaluation results can be found at `http://www.vis.uni-stuttgart.de/~thomds/iddeval/`

References

Ahern, S., Naaman, M., Nair, R., and Yang, J. H.-I. (2007). World Explorer: Visualizing aggregate data from unstructured text in geo-referenced collections. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, JCDL '07, pages 1–10, New York, NY, USA. ACM.

Bosch, H., Thom, D., Wörner, M., Koch, S., Püttmann, E., Jäckle, D., and Ertl, T. (2011). ScatterBlogs: Geo-spatial document analysis. In *Visual Analytics Science and Technology, 2011. VAST 2011. IEEE Conference on*.

Chew, C. and Eysenbach, G. (2010). Pandemics in the age of : Content analysis of tweets during the 2009 H1N1 outbreak. *PLoS One*, 5(11).

Eisenstein, J., O'Connor, B., Smith, N. A., and Xing, E. P. (2010). A latent variable model for geographic lexical variation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 1277–1287, Stroudsburg, PA, USA. Association for Computational Linguistics.

Finkel, R. and Bentley, J. (1974). Quad trees — a data structure for retrieval on composite keys. *Acta informatica*, 4(1):1–9.

Heverin, T. and Zach, L. (2010). Microblogging for crisis communication: Examination of Twitter use in response to a 2009 violent crisis in seattle-tacoma, washington area. In *Proceedings of the 7th International ISCRAM Conference–Seattle*.

Hughes, A. and Palen, L. (2009). Twitter adoption and use in mass convergence and emergency events. *International Journal of Emergency Management*, 6(3):248–260.

Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

MacEachren, A., Jaiswal, A., Robinson, A. C., Pezanowski, S., Savelyev, A., Mitra, P., Zhang, X., and Blanford, J. (2011). SensePlace2: GeoTwitter analytics support for situational awareness. Providence, RI. IEEE Conference on Visual Analytics Science and Technology.

Manning, C. D., Raghavan, P., and Schtze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.

Marcus, A., Bernstein, M., Badar, O., Karger, D., Madden, S., and Miller, R. (2011). TwitInfo: Aggregating and visualizing microblogs for event exploration. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, pages 227–236. ACM.

Mendoza, M., Poblete, B., and Castillo, C. (2010). Twitter Under Crisis: Can we trust what we RT? In *Proceedings of the First Workshop on Social Media Analytics*, pages 71–79. ACM.

Palen, L., Vieweg, S., Liu, S., and Hughes, A. (2009). Crisis in a networked world: features of computer-mediated communication in the april 16, 2007, virginia tech event. *Social Science Computer Review*.

Parzen, E. (1962). On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076.

Roller, S., Speriosu, M., Rallapalli, S., Wing, B., and Baldridge, J. (2012). Supervised text-based geolocation using language models on an adaptive grid. In *EMNLP-CoNLL*, pages 1500–1510. ACL.

Rosenblatt, M. (1956). Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27(3):832–837.

Sakaki, T., Okazaki, M., and Matsuo, Y. (2010). Earthquake shakes Twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on world wide web*, pages 851–860. ACM.

Serdyukov, P., Murdock, V., and van Zwol, R. (2009). Placing flickr photos on a map. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 484–491, New York, NY, USA. ACM.

Sinnott, R. W. (1984). Virtues of the Haversine. *Sky and Telescope*, 68(2):159+.

Starbird, K. and Palen, L. (2010). Pass it on?: Retweeting in mass emergency. In *Proceedings of the 7th International ISCRAM Conference, Seattle, WA*.

Thom, D., Bosch, H., Koch, S., Wörner, M., and Ertl, T. (2012). Spatiotemporal anomaly detection through visual analysis of geolocated Twitter messages. In *IEEE Pacific Visualization Symposium*.

Westover, L. A. (1991). *Splatting: a parallel, feed-forward volume rendering algorithm*. PhD thesis, Chapel Hill, NC, USA. UMI Order No. GAX92-08005.

Wing, B. P. and Baldridge, J. (2011). Simple supervised document geolocation with geodesic grids. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 955–964, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zhang, D., Ooi, B., and Tung, A. (2010). Locating mapped resources in web 2.0. In *Data Engineering (ICDE), 2010 IEEE 26th International Conference on*, pages 521–532. IEEE.