Mining and Managing User-Generated Content and Preferences

Georgios Valkanas*

National and Kapodistrian University of Athens
Department of Informatics and Telecommunications
gvalk@di.uoa.gr

Abstract. Capturing users preferences in today's web ecosystem is essential to provide engaging services. In this thesis, we propose various ways that take user preferences into account. Skyline queries are a characteristic example. However, a problem is that the result size may be too large. To address this issue we propose techniques to diversify it, which is an NP-Hard problem and propose efficient approximate solutions. Alternatively, users describe their experiences with a product or service, and what aspects they liked (or did not) in online reviews. Using this information, we propose techniques to identify competitive products. We present a formal framework for the identification of competitors, which we evaluate extensively and demonstrated its efficiency and efficacy. Finally, users post their preferences and interests online, in social media platforms. Social media data can be used to identify events that occur in the physical world. However, given the domain's particularities, straightforward solutions do not perform well. Therefore, we resort to affective theories of emotion from psychology. Using these theories as a starting point, we monitor the aggregate emotional state of large, geographical groups of people and automatically identify abrupt changes, linking them to the underlying events. We develop *i*) custom geocoding techniques, *ii*) a classification framework mapping social data to emotions, *iii*) an online outlier detection algorithm to identify abrupt changes, and *iv*) a visual component to ease the presentation of events.

Keywords: web mining, review mining, event detection, user preferences

1 Dissertation Summary

The World Wide Web has changed dramatically over the years since its initial inception, and is still evolving as new technologies emerge. Online services and applications are more pervasive nowadays, allowing users to share online aspects of their everyday lives. More importantly, users feel *comfortable* with doing so, which is a major shift in their attitude regarding privacy in digital environments. This general change in behavior has made the boundaries between the physical and online world less transparent.

* Dissertation Advisor: Dimitrios Gunopulos, Professor

Sharing of information takes place in various forms. The common denominator is that users express their preferences and personal opinions on various topics, such as music, products, politics, etc. Although new technologies provide the necessary framework(s) for the users to express themselves, novel techniques are required to turn the available data into useful and actionable information. Such a need translates into interesting and challenging research questions, which we have to address, in order to provide the next generation services. For instance, more expressive query types are needed, whereby user preferences can be taken into account. At the same time, we should develop techniques that extract meaningful and insightful information from this high-volume, user-generated content.

Skyline queries are an indicative example. These queries support multi-objective optimization and are geared towards returning items with different trade-offs. Although easier to understand (from the user's perspective), given that preferences are defined on each attribute separately, the output size can become extremely large. It would, then, be very tedious for the user to inspect all the results manually and soem post-processing of the results is necessary.

One way to address this problem is to incorporate additional constraints in the result selection process. For example, we can select a subset of skyline points that meet certain criteria or that optimize a given objective function. *Diversifying* the skyline result is one such approach, which is also highly desirable considering that *skyline* queries aim to return points with trade-offs [26]. Selecting a subset of the skyline points has been looked into in the past, with different goals in mind. In particular, the work in [12] proposed the selection of k skyline points that maximize the coverage of the space. Another approach selected k points that best describe the skyline contour [17], which could be seen as a diversification approach of the skyline result. However, in this thesis we demonstrated that coverage solves a *different* problem from *diversification*. Moreover, the technique in [17] assumes an underlying Euclidean space, which is not always the case for skyline queries. Therefore, that technique becomes unusable in non-Euclidean spaces, in partially-ordered domains or in feature spaces with non numerical attributes, where skyline queries are still meaningful. In addition, the result of [17] may change depending on the weights of each feature, which negates the axis-weighting invariance property of skyline queries. Finally, as we demonstrated in our thesis, the subset of skyline points selected by [17] correlate more with coverage than with diversity.

Ranking the skyline points could be another approach, giving them a degree of importance and returning them in an ordered fashion. Ranking the skyline has been the focus of past research [29, 6, 31]. However, these techniques face one of two problems: *i*) they may return points that are **not** part of the skyline [31] or *ii*) they return skyline points with *extreme* values in a single dimension [29, 6]. The latter techniques also consider an exponentially large search space, given that they take into account all non-empty feature subspaces (at most $O(2^d)$, where d is the number of considered dimensions). To counter these problems, we propose to rank skyline points using a novel technique, inspired by Information Retrieval. In particular, we adapt the well known Term Frequency-Inverse Doc-

ument Frequency (TF-IDF) scheme to the skyline domain, and propose efficient techniques to rank skyline points accordingly [27].

User feedback can be provided in other formats as well, such as semi-structured and free text. Online reviews fall under the second category, and has received considerable attention in recent years. This increased attention is due to the impact that reviews have in the marketability of products. In fact, surveys have shown that users prefer products that have already been reviewed, so that they know the item's pros and cons, and can, therefore, make informed decisions. Through a combination of user feedback and product specifications, we can derive a rigid framework to analyze and compare such products.

More specifically, based on the users' needs - as expressed in their feedback - and the extent to which a product can cover similar needs - given by its characteristics -, we can identify how *competitive* two products are. This is extremely useful for both item producers (e.g., the companies), as well as item consumers (e.g., the end users). Despite its importance, a formal framework to identify *competitive* items had been largely missing until now. The recent availability of online reviews has allowed us to test both the efficiency and efficacy of techniques that return the top-k *most competitive* products, with respect to a given item of interest [10].

Previous works on *competitor* identification has focused on the retrieval of comparative expressions, such as "X is better / worse than Y", or "(product) vs (product) Y" [2, 11]. The underlying assumption is that if two products co-occur frequently in such expressions, they are more competitive, as opposed to products that occur less frequently. The problem with this approach is that, oftentimes, there is a scarcity of such expressions, making our confidence in the drawn results quite weak. Additionally, these approaches are only useful when a product is compared as a whole, whereas users may discuss certain features of a product in their reviews. For these reasons, in this thesis we propose a rigid framework, utilizing both product specifications as well as user feedback at the feature-level. Competitiveness is then measured as the degree to which two products fulfill the same needs of groups of people with similar requirements. We present techniques to efficiently retrieve the top-k competitors of a given item, and evaluate our method's efficacy using a user study. Our results demonstrate that our techniques are very efficient, and that our model aligns well with users' intuition of competitiveness.

Finally, despite the sharp increase in numbers of online reviews over the years, these are nowhere near the data volume produced in social media. Popular social media platforms have extremely high user adoption, with Facebook boasting more than 1.28 *billion* active users per month (as of March 31, 2014), and Twitter - a later founded company - having more than 255 million active users per month (as of July 2014). A driving force of these frameworks is their networking component, with people linking to one another, as a prerequisite to share information. Undoubtedly, social media is among the most prolific areas for research nowadays, not only because of the user adoption, but also due to the usefulness of the data in various diverse disciplines: computer science, psy-

chology, sociology and journalism to name a few. Moreover, there are practical applications where the data can be used. Advertising and community detection are typical use cases, whereas (real-time) event detection, interaction analysis, and user behavior understanding increasingly gain attention. Making sense of the user-genereated content in these mediums is also extremely challenging, because of the data volume and content diversity, which is as high as the underlying population and their interests.

As a first objective, we wanted to explore the properties of data posted in social media platforms, to better understand the kind of information we are dealing with. Towards this direction, a major outcome of this thesis is to show that elevated access is primarily needed for applications that rely heavily on up-to-date information and do not only focus on popular items. Applications that only deal with popular items, can be well served through default access [24].

The fast pace of social media platforms is a key factor in considering them as online news reporting tools. However, mining high volumes of data to identify (newsworthy) events is far from trivial. Previous techniques have focused on event monitoring, implying that the event is already identified or somehow known [13]. Others simplify the problem by searching for specific keywords, which can accurately describe the event [16]. Online clustering techniques have also been explored [3, 30], however, they do not perform well in fast-paced mediums [20]. It is easy to see that identifying events, regardless of type and without prior knowledge of any descriptive keywords, calls for a different approach.

For this reason, we resort to psychological theories, according to which events impact the user psychologically, and more specifically their affective state, compelling them to externalize their thoughts. We argue that newsorthy events will impact large groups of users, and by monitoring a group's aggregate affective / emotional state, we will be able to capture abrupt changes and trace them back to the source, i.e., the event.

Within this research question, however, there are several other issues to resolve. In particular, we must identify an event's location, so we develop custom geocoding techniques, that convert textual information to GPS coordinates [18]. Extracting the affective state of a single user is challenging on its own, let alone for an entire group. We solve this problem through a classification framework, mapping social media data to a set of predefined basic emotions [19]. Capturing abrupt changes requires a careful formulation of the problem, as well as efficient computation techniques, due to the high volumes of real-time data we are dealing with. We formulate this problem as an instance of online outlier detection and propose online techniques that approximate the Probability Density Function (PDF) of the aggregate emotional state [20]. Information visualization is also important in that domain, to better explain an occuring event. Therefore, we propose a User Interface that presents all of the information in an appropriate way [21]. Such an approach also requires a great deal of system and software engineering, and end-to-end solutions could also be used to facilitate the data harvesting process [25, 23, 28].

2 Results and Discussion

2.1 Skyline Diversification

Let $\mathcal{D}$ be a d-dimensional dataset, where w.l.o.g. smaller values are preferred, i.e., we are interested in *minimizing* each attribute. [1] We say that $p = (p.x_1, ..., p.x_d) \in \mathcal{D}$ *dominates* $q = (q.x_1, ..., q.x_d) \in \mathcal{D}$ (and write $p \prec q$), when: $\forall i \in \{1, ..., d\}, p.x_i \leq q.x_i \wedge \exists j \in \{1, ..., d\} : p.x_j < q.x_j$. The skyline $\mathcal{S} \subseteq \mathcal{D}$, is composed of all points in $\mathcal{D}$ that are not dominated by any other point.

To overcome the limitations of a Euclidean space assumption, we propose to use the *Jaccard distance* for diversity computation. Each skyline point p is associated with the set of points that it dominates, denoted by $\Gamma(p) = \{q \in \mathcal{D} | p \prec q\}$. The *domination score* of p is the cardinality of $\Gamma(p)$. The similarity between p and q is defined as the Jaccard similarity between the sets $\Gamma(p)$ and $\Gamma(q)$, i.e.,

$$J_s(p, q) = \frac{|\Gamma(p) \cap \Gamma(q)|}{|\Gamma(p) \cup \Gamma(q)|}$$

and ranges between 0 and 1. The corresponding distance measure is thus $J_d(p, q) = 1 - J_s(p, q)$ and it is well known that it satisfies all metric properties. We select the Jaccard distance as a measure of diversity because:

i) it relies solely on the dominance relations among points, therefore, no user-defined distance function or other input is required,
ii) the quality of the resulting set of points does not depend on the skyline $\mathcal{S}$ alone, but on the characteristics of $\mathcal{D}$ as well
iii) it leads to elegant ways of diversity computation by means of min-wise independent permutations, and
iv) it is the most widely accepted measure for set (dis)similarity.

We model k-diversity as a *k-dispersion* problem, which is NP-Hard [9]. In k-dispersion, the goal is to find k objects that optimize an objective function of their distance. The optimal solution is given by:

$$OPT = \arg \max_{\substack{\mathcal{A} \subseteq \mathcal{S} \\ |\mathcal{A}| = k}} f(\mathcal{A})$$

There are two basic alternatives for the objective function: *i*) maximize the sum of distances (k-MSDP) and *ii*) maximize the minimum distance (k-MMDP). Although both alternatives are valid, we choose to work with k-MMDP because *i*) it leads to 2-approximation algorithms, instead of the 4-approximation of k-MSDP [15], and *ii*) it intuitively returns results of better quality.Given the NP-Hardness of the problem, we resort to approximate solutions. In fact, the greedy approach can be quite inefficient, due to a large number of range queries. Therefore, we propose the use of the MinHashing technique [5], that transforms

[1] We focus on numerical attributes for ease of presentation. Our approach applies to categorical ones equally well.

the original space into a more compact one, where computations are much faster. In particular, we develop the SkyDiver framework, which operates in two phases.

Phase 1: Fingerprinting. This phase generates a *signature* of reduced size for each skyline point, based on MinHashing. Alternatively, we can use *Locality Sensitive Hashing* (LSH) as a memory efficient alternative.

Phase 2: Selection. This phase is responsible for selecting the k most diverse skyline points, and can be applied to either the MinHash or the LSH signatures.

Assume that the data set is viewed as a matrix M with $n - m$ rows and m columns, $m = |\mathcal{S}|$ and $n = |\mathcal{D}|$. Each skyline point is represented by a single column, whereas a dominated point is represented by a row. In this matrix, $M[i, j] = 1$ iff the j-th skyline point dominates the i-th data point and 0 otherwise. Let $\mathcal{H} = \{h_1, ..., h_t\}$ be a set of t min-wise independent hash functions, where each h_i performs a random permutation of the rows. The cardinality of $\mathcal{H}$ (i.e., the number of hash functions used) determines the size of each signature. To generate random permutations, each hash function $h_i \in \mathcal{H}$ is of the form

$$h_i(x) = a_i \cdot x + b_i \mod P$$

where P is a prime number larger than $n - m$ and a_i, b_i are randomly chosen constants taking integer values in $[1, P]$. According to [5], if $J_s(p, q)$ is the Jaccard similarity between skyline points p and q, then for each hash function h_i it holds

$$Prob[h_i(p) = h_i(q)] = J_s(p, q).$$

Recall that each row of the matrix M is a bit-array. If $M[i, j] = 1$ then the $\mathcal{S}_j \prec \mathcal{D}_i$. Each row is hashed t times, by every $h_i \in \mathcal{H}$ and the signature of each skyline point is updated accordingly. Therefore, each signature is composed of t integer values. According to [7], if $\Omega(\varepsilon^{-3}\beta^{-1}\log(1/\delta))$ is the signature size, where ε is the maximum allowed error ($0 < \varepsilon < 1$), then with probability at least $1 - \delta$ it holds

$$(1 - \varepsilon)J_s(p, q) + \varepsilon\beta \leq \widehat{J}_s(p, q) \leq (1 + \varepsilon)J_s(p, q) + \varepsilon\beta$$

where $0 < \beta < 1$ is the required precision.

Given the signature matrix $\widehat{M}$, we can select the k skyline points in the transformed space, which is a metric space. Therefore we can use the greedy approach on the signatures and acquire a 2-approximation solution. However, due to distance distortions, as a result of embedding the distances in lower dimensionality (through MinHashing), it is possible to obtain a sub-optimal solution. The following theorem relates the true optimal solution, to the one computed by working with MinHash signatures.

Theorem 1. *Let OPT be the value of the optimal solution to the k-diversity problem in the original space and let x, y denote the corresponding skyline points, i.e., $J_d(x, y) = OPT$. Similarly, let $\widehat{OPT}$ be the optimal value if the problem is solved using MinHash signatures and let a, b be the corresponding skyline points, i.e., $\widehat{J}_d(a, b) = \widehat{OPT}$. For a given ε and sufficiently small δ, it holds that: $J_d(a, b) \geq \frac{1+\varepsilon}{1-\varepsilon}OPT - \frac{2\varepsilon}{1-\varepsilon}$.*

2.2 Competitor Mining

Competitiveness is a challenge that every product or service provider has to face, regardless of the application domain. A significant amount of relevant work has demonstrated the strategic importance of identifying and monitoring an entity's competitors [14]. In this thesis we focus on a formal framework for competitor identification:

Problem 1. We are given a set of items $\mathcal{I}$, defined within the feature space $\mathcal{F}$ of a particular domain. Then, given any pair of items I, I' from $\mathcal{I}$ we want to define a function $C_{\mathcal{F}}(I, I')$ that computes the competitiveness between the two in the context of the domain.

Figure 1 provides a (simplified) overview of our approach, where we illustrate the competitiveness between three different items I_1, I_2 and I_3. Each item is mapped to the set of features that it can offer to the users. Three distinct features are considered in this example: A, B and C. Note that, for this simple example, we only consider binary features (i.e. available/not available). Our actual formalization accounts for a much richer space of binary, categorical and numerical features. The left side of the figure shows three groups of users (g_1, g_2, g_3). The example assumes that these are the only groups in existence. Users are grouped based on their preferences with respect to the features. For example, the users in group g_2 are only interested in features A and B. As can be seen by the figure, items I_1 and I_3 are not competitive to each other, since they simply do not appeal to the same groups of users. On the other hand, I_2 is in competition with both I_1 (for groups g_1 and g_2) and I_3 (for g_3). Finally, another interesting observation is that I_2 competes with I_1 for a total of 4 users, and with I_3 for a total of 9 users. In other words, I_3 is a stronger competitor for I_2, since it claims a much larger portion of I_2's market-share than I_1. In our work, we propose ways to deduce these user-groups from sources such as query logs and customer reviews, and describe methods to estimate the size of the market

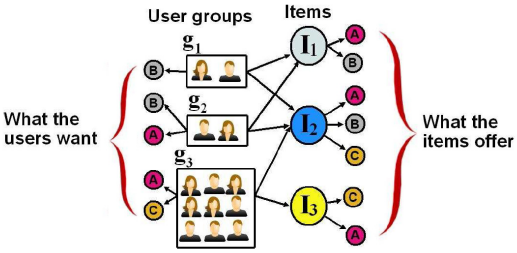


Fig. 1. Simplified example of our competitiveness paradigm

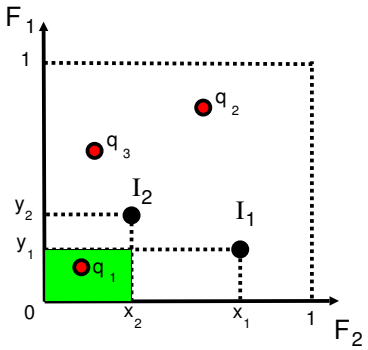


Fig. 2. Geometric interpretation of pairwise coverage

share that they represent. Our work is the first to utilize the opinions expressed in customer reviews as a resource for mining competitiveness.

In order to evaluate the competitiveness of two given items I_i, I_j in the context of a subset of features $\mathcal{F}'$, we need to compute the number of possible value assignments over $\mathcal{F}'$ that are satisfied by *both* items. Formally, we define pairwise coverage as follows:

Definition 1. [Pairwise Coverage] *Given the complete set of features $\mathcal{F}$ in a domain of interest, let $\mathcal{V}_{\mathcal{F}'}$ be the complete space of all possible value-assignments over the features in a subset $\mathcal{F}' \subseteq \mathcal{F}$. Then, the* coverage $cov(\mathcal{V}_{\mathcal{F}'}, I_i, I_k)$ *of a pair of items I_i and I_j with respect to $\mathcal{V}_{\mathcal{F}'}$ is defined as the portion of $\mathcal{V}_{\mathcal{F}'}$ that is covered by both items.*

Considering the above definition, we observe that the coverage of each dimension (i.e. each feature $F \in \mathcal{F}'$) is independent of the others. Therefore, we first compute the percentage of each dimension that is covered by the pair. We can then optimally compute the coverage of the entire space $\mathcal{V}_{\mathcal{F}'}$ as the product of the respective coverage values $\mathcal{V}_{\{F\}}$ for every $F \in \mathcal{F}'$. Formally:

$$cov(\mathcal{V}_{\mathcal{F}'}, I_i, I_j) = \prod_{F \in \mathcal{F}'} cov(\mathcal{V}_{\{F\}}, I_i, I_j) \tag{1}$$

This computation has a clear geometric interpretation: The portion of the space $\mathcal{V}_{\mathcal{F}'}$ that is covered by a pair of items can be represented as a hyper-rectangle in $|\mathcal{F}'|$-dimensional space. For each dimension F, $cov(\mathcal{V}_{\{F\}}, I_i, I_j)$ gives us the portion of the dimension that is covered by the two items. Finally, by multiplying the individual coverage values, we are essentially computing the volume of the hyper-rectangle that represents the entire space $\mathcal{V}_{\mathcal{F}'}$. This is graphically portrayed in Figure 2, which shows the common coverage of two items I_1, I_2 (green area) in the context of a dimensional space $\{F_1, F_2\}$.

Definition 1 allows us to evaluate the coverage provided by a pair of items to (the value space of) any subset of features $\mathcal{F}'$. Conceptually, $\mathcal{F}'$ captures the fraction of the population that is interested in the features included in $\mathcal{F}'$. Further, we define $\mathcal{Q}$ to be the collection of subsets with a non-zero weight. Formally: $\mathcal{Q} = \{\mathcal{F}' \in 2^{\mathcal{F}} : w(\mathcal{F}') > 0\}$. Taking the above into consideration, we formally define the competitiveness of two items I_i, I_i as follows:

Definition 2. [Competitiveness] *Given the complete set of features $\mathcal{F}$ of a domain of interest, let $\mathcal{Q}$ be the set of all subsets of $\mathcal{F}$ that have a non-zero popularity weight. Then, the competitiveness of two given items I_i and I_i is defined as:*

$$C_{\mathcal{F}}(I_i, I_j) = \sum_{\mathcal{F}' \in \mathcal{Q}} w(\mathcal{F}') \times cov(\mathcal{V}_{\mathcal{F}'}, I_i, I_j) \tag{2}$$

where $cov(\mathcal{V}_{\mathcal{F}'}, I_i, I_j)$ is the portion of $\mathcal{V}_{\mathcal{F}'}$ that is covered by both I_i and I_j.

Given this definition of competitiveness, we study the natural problem of finding the top-k competitors of a given item I^*:

Problem 2. We are given a set of items $\mathcal{I}$, defined within the feature space $\mathcal{F}$ of a domain. Then, given a single item $I \in \mathcal{I}$, we want to identify the k items from $\mathcal{I} \setminus \{I\}$, that maximize the pairwise competitiveness with I:

$$I^* = \underset{I' \in \mathcal{I} \setminus \{I\}}{\operatorname{argmax}} C_{\mathcal{F}}(I, I') \tag{3}$$

Instead of finding the top-k competitors using a naive solution that iterates over all items, computes their competitiveness with respect to I^* and finally orders them, we develop a more efficient technique, namely *CMiner*. Our algorithm makes use of an indexing scheme, called the *Dominance Pyramid*, which is built based on the dominance property of items, as discussed in the skyline section. It also applies very efficient pruning on the search space, by bounding the score of candidate points. Building upon a result from [4], in this thesis, we show that the algorithmic complexity of our technique is O($|\mathcal{I}|$ * $|\mathcal{Q}|$ * k^2), where $\mathcal{Q}$ is the set of feature subsets with non-zero weights [2].

2.3 Event Detection

Our objective with social media data [3] can be summarized as follows:

Problem 3. **[Event Detection]** Given a time ordered stream of tweets, identify those posts which *i*) alter significantly and abruptly the emotional state of a (potentially) large group of users, and *ii*) can be traced back to event e.

Event Extraction Workflow The problem we described fits nicely with an outlier detection definition. Figure 3 shows a schematic overview of our system [4]

[2] This is to retrieve the top-k competitors of every item in the dataset
[3] We focus on Twitter, http://twitter.com
[4] Storage image by Barry Mieny, under CC BY-NC-SA license.

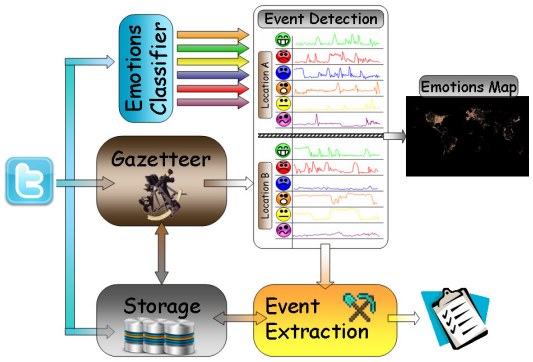


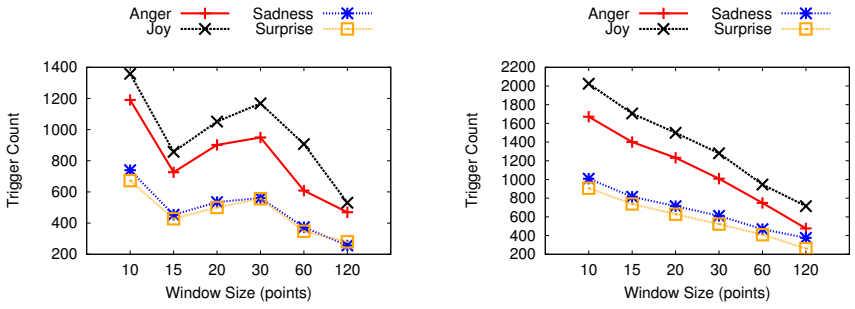
Fig. 3. Schematic interaction of our system's components

used to identify events. The Twitter stream is our input, feeding two components, namely the *emotions classifier* and the *location extraction* subsystem. Through a custom geocoding component [18], each incoming tweet is mapped to a GPS location, which will also be the location of the event (assuming one occurs).

We classify tweets to 6 basic emotions, proposed by American psychologist Paul Ekman [8]: *anger, fear, disgust, happiness, sadness, surprise*, plus a neutral one (*none*), to describe the absence of an emotion. Tweets with non-neutral emotions are further processed by virtual sensors, one per location and per emotion. Virtual sensors count how many tweets they have received during the last aggregation interval a (system parameter). These values form the aggregate emotional state, which are fed to the event detection mechanism. By operating over the w most recent values, the event detection module approximates their distribution using non-parametric models (kernel estimators in particular), estimating the Probability Density Function (*PDF*). The same models are used for event detection, identifying tuples which are outside of the norm over the most recent history of tuples, given by the combination of (a, w) parameters.

The Temporal Nature of Twitter Discussions Using a large crawl of Twitter data (2 month period), we evaluated the behavior of the medium and our event extraction approach. In particular, we computed the number of times a "trigger" was raised (which corresponds to an individual event in our case). Figure 4 shows the number of times our approach raised an event as a function of the history it maintains, when aggregating emotions over the past 1 minute and monitoring the entire stream at once (we use only one sensor).

Interestingly, a bigger sample size results in more triggers. This is due to maintaining outdated information compared to the fast pace of the medium. Increasing the history length results in fewer triggers (Figure 4(b)), because new points can be matched against more sampled data, and are less likely to be flagged as outliers. On the other hand, Figure 4(a) leads to a very interesting observation. Using a 50% sample, there is a dramatic drop in the number of triggers, when we increase the window size from 10 to 15; from that point, until



(a) Sample size = 50%

(b) Sample size = 100%

Fig. 4. #Times a trigger was raised, w.r.t. the window size. a = 1min, r = 0.01, p=0.1

a window size of 30, triggering events increases slightly, and begins decreasing from that point on. This means that for 1 minute aggregations, there are rapid changes in the observed emotions; therefore a window of 10 points may be too narrow, to maintain a representative "history". On the other hand, a window between 15 – 30 minutes seems like a better choice. This result correlates very well with the real time nature of the medium, where people tend to speak and respond very quickly to their tweets. It also means that events that are present in our data create some momentum over a mid-size period (∼30minutes), and then dissipate.

3 Conclusions

In this thesis, we considered various scenarios where user preferences can be taken into account, in order to improve the quality of a provided service. In particular, we focused on different use cases, where user preferences play a vital role, namely: *i*) skyline queries, *ii*) review mining and competitor identification, and *iii*) social media. The problems that arise in each of these domains are unique, and we proposed techniques to efficiently solve them, while providing quality guarantees on our solutions. Our analysis also revealed some interesting properties for the social media domain. Finally, we effectively modeled competitors in a formal framework. User preferences and feedback may come in different forms, such as mouse movements [1], or interaction with online content [22].

References

1. I. Arapakis, M. Lalmas, and G. Valkanas. Understanding within-content engagement through pattern analysis of mouse gestures. In *CIKM*, pages 1439–1448, 2014.
2. S. Bao, R. Li, Y. Yu, and Y. Cao. Competitor mining with the web. *IEEE Transactions on Knowledge Data Engineering*, pages 1297–1310, 2008.
3. H. Becker, M. Naaman, and L. Gravano. Learning similarity metrics for event identification in social media. In *WSDM*, pages 291–300, 2010.
4. J. L. Bentley, H. T. Kung, M. Schkolnick, and C. D. Thompson. On the average number of maxima in a set of vectors and applications. *Journal of the ACM*, 25(4):536–543, 1978.
5. A. Z. Broder, M. Charikar, A. M. Frieze, and M. Mitzenmacher. Min-wise independent permutations. *Journal of Computer and System Sciences*, 60(3):630–659, 2000.
6. C. Y. Chan, H. V. Jagadish, K.-L. Tan, A. K. H. Tung, and Z. Zhang. On high dimensional skylines. In *EDBT*, pages 478–495, 2006.
7. M. Datar and S. Muthukrishnan. Estimating rarity and similarity over data stream windows. In *ESA*, pages 323–334, 2002.
8. P. Ekman, W. Friesen, and P. Ellsworth. *Emotion in the human face: guide-lines for research and an integration of findings*. Pergamon Press, 1972.
9. C.-C. Kuo, F. Glover, and K. S. Dhir. Analyzing and modeling the maximum diversity problem by zero-one programming. *Decision Sciences*, 24(6):1171–1185, 1993.

10. T. Lappas, G. Valkanas, and D. Gunopulos. Efficient and domain-invariant competitor mining. In *SIGKDD*, pages 408–416, 2012.
11. R. Li, S. Bao, J. Wang, Y. Liu, and Y. Yu. Web scale competitor discovery using mutual information. In *ADMA*, 2006.
12. X. Lin, Y. Yuan, Q. Zhang, and Y. Zhang. Selecting stars: The k most representative skyline operator. In *ICDE*, pages 86–95, 2007.
13. M. Mathioudakis and N. Koudas. Twittermonitor: trend detection over the twitter stream. In *SIGMOD*, 2010.
14. M. E. Porter. *Competitive Strategy: Techniques for Analyzing Industries and Competitors*. Free Press, 1980.
15. S. S. Ravi, D. J. Rosenkrantz, and G. K. Tavyi. Heuristic and special case algorithms for dispersion problema. *Operations Research*, 42(2):299–310, 1994.
16. T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *WWW*, pages 851–860, 2010.
17. Y. Tao, L. Ding, X. Lin, and J. Pei. Distance-based representative skyline. In *ICDE*, pages 892–903, 2009.
18. G. Valkanas and D. Gunopulos. Location extraction from social networks with commodity software and online data. In *ICDM Workshops (SSTDM)*, 2012.
19. G. Valkanas and D. Gunopulos. Event detection from social media data. *IEEE Data Eng. Bull.*, 36(3):51–58, 2013.
20. G. Valkanas and D. Gunopulos. How the live web feels about events. In *CIKM*, 2013.
21. G. Valkanas and D. Gunopulos. A ui prototype for emotion-based event detection in the live web. In *SS-KDD-HCI @ SouthCHI*, pages 89–100, 2013.
22. G. Valkanas and D. Gunopulos. Predicting download directories for web resources. In *WIMS*, page 8, 2014.
23. G. Valkanas, D. Gunopulos, I. Galpin, A. J. G. Gray, and A. A. A. Fernandes. Extending query languages for in-network query processing. In *MobiDE*, pages 34–41, 2011.
24. G. Valkanas, I. Katakis, D. Gunopulos, and A. Stefanidis. Mining twitter data with resource constraints. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 182–187, 2014.
25. G. Valkanas, A. Ntoulas, and D. Gunopulos. Rank-aware crawling of hidden web sites. In *WebDB*, 2011.
26. G. Valkanas, A. N. Papadopoulos, and D. Gunopulos. Skydiver: A framework for efficient skyline diversification. In *EDBT*, pages 406–417, 2013.
27. G. Valkanas, A. N. Papadopoulos, and D. Gunopulos. Skyline ranking à la IR. In *ExploreDB*, pages 182–187, 2014.
28. G. Valkanas, A. Saravanou, and D. Gunopulos. A faceted crawler for the twitter service. In *WISE*, pages 178–188, 2014.
29. A. Vlachou and M. Vazirgiannis. Ranking the sky: Discovering the importance of skyline points through subspace dominance relationships. *Data & Knowledge Engineering*, 69(9):943–964, 2010.
30. J. Weng and B.-S. Lee. Event detection in twitter. In *ICWSM*, 2011.
31. M. L. Yiu and N. Mamoulis. Efficient processing of top-k dominating queries on multi-dimensional data. In *VLDB*, pages 483–494, 2007.