Detecting Anomalies in Dynamic Rating Data: A Robust Probabilistic Model for Rating Evolution

Stephan Günnemann Nikou Günnemann Christos Faloutsos
Carnegie Mellon University, USA
{sguennem, nguennem, christos}@cs.cmu.edu

ABSTRACT

Rating data is ubiquitous on websites such as Amazon, TripAdvisor, or Yelp. Since ratings are not static but given at various points in time, a temporal analysis of rating data provides deeper insights into the evolution of a product's quality. In this work, we tackle the following question: Given the time stamped rating data for a product or service, how can we detect the general rating behavior of users as well as time intervals where the ratings behave anomalous?

We propose a Bayesian model that represents the rating data as sequence of categorical mixture models. In contrast to existing methods, our method does not require any aggregation of the input but it operates on the original time stamped data. To capture the dynamic effects of the ratings, the categorical mixtures are temporally constrained: Anomalies can occur in specific time intervals only and the general rating behavior should evolve smoothly over time. Our method automatically determines the intervals where anomalies occur, and it captures the temporal effects of the general behavior by using a state space model on the natural parameters of the categorical distributions. For learning our model, we propose an efficient algorithm combining principles from variational inference and dynamic programming. In our experimental study we show the effectiveness of our method and we present interesting discoveries on multiple real world datasets.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications —*Data mining*; I.2.6 [**Artificial Intelligence**]: Learning

Keywords

robust mining; anomaly detection; categorical mixtures

1. INTRODUCTION

Online rating data provides customers valuable information about products and services and supports their decision making process. Exploiting and presenting this data is a key feature of websites such as Amazon, Yelp, or Tripadvisor. Besides the usefulness of rating data for customers, also companies and manufactures can benefit from it by, e.g., using the data to detect functional weaknesses of their products or changes in the customers' satisfaction.

In this work, we propose a method for analyzing rating data that incorporates the data's temporal characteristics. Given the time stamped rating data for a product or service, we aim at detecting the general rating behavior of the users (called the base behavior) as well as time intervals in which these ratings deviate from the general population (anomalous behavior). The base behavior describes the general quality of a product or service accounting for temporal evolutions, e.g., resulting from decreasing quality due to technical progress of competing products. The anomalies, in contrast, deviate from the base behavior and might, e.g., occur due to spammers trying to push the success of a product or due to changes in the service quality.

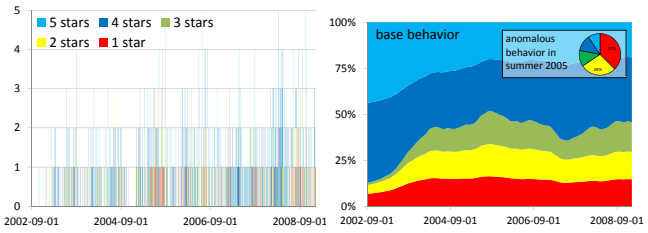

Fig. 1: Left: Time-stamped rating data analyzed by our method (here: a hotel from TripAdvisor). Right: Extracted base behavior. Upper corner: Anomalous behavior detected in summer 2005.

In Figure 1, we show a real world example for such an effect. The data we analyzed here is a hotel from the TripAdvisor database. On the left, we show a subset of the original time stamped data. The colors indicate the different star ratings, and the height of each bar the number of these ratings at the current time. Obviously, it is hard to analyze such data by hand. In particular, keep in mind that the ratings are not uniformly distributed over time.

On the right, we illustrate the detected *base behavior* of our method. As one can see, the base behavior nicely shows the general rating behavior of the users and evolves smoothly over time with primarily medium to high ratings. Additionally, our method has found anomalous behavior in the months of July and August 2005. As shown on the upper right, in these intervals, the fraction of low ratings (red and yellow) is highly increased compared to the base behavior (65% low ratings compared to around 30% in the base behavior). As we will show in Section 5, these anomalies occurred due to problems in the service of the hotel.

KDD'14, August 24–27, 2014, New York, NY, USA.

http://dx.doi.org/10.1145/2623330.2623721.

In general, our method detects time intervals in dynamic rating data which show anomalous behavior and – at the same time – it detects the base behavior if the data would *not* be polluted by anomalies. Besides using this principle to detect weaknesses in products and services, it can generally be used to filter out rating information which behave anomalous. Thereby, prospective customers might be provided with a cleaned history about the product; or, one might specifically highlight these time intervals to the users to provide the whole picture on a product (since otherwise these anomalous ratings are hidden in the larger set of normal behavior). As an additional benefit, we can exploit our method to predict the base behavior of future ratings. Accordingly, when new ratings arrive, we can estimate whether they match or deviate from the predicted behavior – thus, giving indication of new anomalies.

So far, there exists only a single method which analyzes temporal rating data under the presence of anomalies [7]. A potential drawback of [7], however, is the necessary aggregation/binning of the data. When using a coarse aggregation, the temporal effects of the data are not well captured (in the extreme, all data is a single bin). When using a fine aggregation, the analyzed distributions might degenerate (in the extreme, many bins are empty). In our model we avoid this problem by directly operating on the time stamped data which is modeled via a sequence of categorical mixture models. We explicitly keep into account that ratings might not uniformly arrive over time. Furthermore, the work [7] assumes that anomalies occur at individual points in time. Our work captures the effects of real data much better by accounting for multiple different types of anomalies appearing across several *time intervals*. Our contributions are:

- *Mining task:* We present a technique for the analysis of time stamped rating data. Our method detects the base behavior of users as well as time intervals where potential anomalies occurred. Additionally, our technique can be used to predict the rating behavior at future time points.
- *Theoretical soundness:* Our method is based on a sound Bayesian model that represents the rating data as a sequence of temporally constrained categorical mixture models. To capture the temporal effects of the base behavior we use a state space model on the natural parameters of the categorical distributions.
- *Algorithm design:* We propose an efficient algorithm for learning our model which combines principles from variational inference and dynamic programming.
- *Effectiveness:* We evaluate our method on different real world datasets and we show its effectiveness by presenting interesting findings.

2. BAYESIAN FRAMEWORK

In this section, we introduce our model for detecting the base rating behavior of users as well as time intervals in which anomalies have been occurred. Following convention, we do not distinguish between a random variable X and its realization $X = x$ if it is clear from the context. As an abbreviation, we denote sets of random variables with the index $*$, e.g. $z_*^{(t)}$ is the set $\{z_i^{(t)}\}$ with i in the corresponding domain, and z is an abbreviation for the set $z_*^{(*)}$. Vectors of (random) variables are written in bold font, e.g. $\boldsymbol{b}$, while the entries of the vectors are given in standard font, e.g. b_i. We write $t \in T$, as a shortcut for $t \in \{1, \ldots, T\}$.

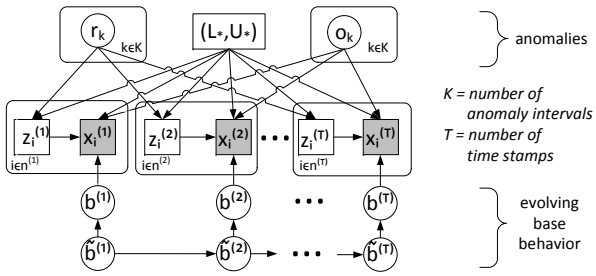

Fig. 2: Graphical model of our approach

Preliminaries. The data we consider is a time stamped collection of ratings. Let $x_i^{(t)}$ denote the i-th rating occurred at *time index* t, and $s^{(t)}$ the *time stamp* at time index t. At each time index we might observe multiple ratings (e.g. if time stamps are only measured/provided on a daily basis). We denote with $n^{(t)}$ the number of ratings at time index t. We assume the data to be ordered according to time, i.e. $s^{(t)}$ occurs after $s^{(t-1)}$. We denote with $\Delta^{(t_1,t_2)}$ the elapsed time between time stamp $s^{(t_1)}$ and $s^{(t_2)}$ and we set $\Delta^{(t)} := \Delta^{(t-1,t)}$. Note that for each t a different $\Delta^{(t)}$ might be observed since we do not require a fixed binning or aggregation of the rating data. We denote with T the number of time indices (i.e. the number of *distinct* time stamps) and we assume that users can choose ratings based on a rating scale with S different ratings (e.g. stars from 1 to S). As an abbreviation for later use, we define $n_s^{(t)} := |\{i \in \{1, \ldots, n^{(t)}\} \mid x_i^{(t)} = s\}|$ to be the number of ratings at time t which possess the evaluation $s \in S$.

Generative Process.

We model the rating data including potential anomalies via a probabilistic generative process. An overview of our generative process showing the used variables and their dependencies is illustrated by the graphical model in Figure 2. In the following we discuss the details of this process.

Given the observed rating data X, our aim is to extract the base behavior of the users and intervals in time where anomalies occur. Since the observed data might already be polluted by anomalies, we cannot directly use it to estimate the base behavior. Instead, we assume that the observed data is obtained by mixing the (unkown) base user behavior with the (unknown) anomaly behavior. Thus, both types of behavior are represented as *latent* variables which are not directly observed but inferred by our technique. Technically, at each point in time we have a categorical mixture model as illustrated in Figure 3. To incorporate the temporal properties of the data, we perform additional modeling:

Part 1: Mixing anomalies and base behavior. In contrast to the base behavior, which is present over the whole timespan, we assume that anomalies are rare events (otherwise, they would correspond to the majority of the data, making the term "anomaly" rather meaningless) and occur only in a specific time interval like, e.g., during a short attack of spam. Technically, instead of using an individual anomaly at each point in time, we restrict the number of anomalies to be small, i.e. smaller than a number K (we discuss later how to choose this parameter), and we temporally constrain the "influence" of each anomaly to a small time interval. For each anomaly, we define this interval by the random variables L_k and U_k, denoting the lower and upper bound of time indices at which the kth anomaly occurs.

In the following, we will use the function

$$k(t) = \begin{cases} k & \text{if } \exists x : x = k \wedge L_x \leq t \leq U_x \\ 0 & \text{else} \end{cases} \quad (1)$$

which maps the time index t to the corresponding anomaly (or to 0 if no anomaly exists at this point in time). Here, we require disjointness of the different intervals.

Note that the use of anomaly *intervals* is a huge advantage in contrast to [7], which models the anomalies at each time point individually. The potential of this extension is best shown for the case of very fine grained time stamps: In this case, we usually expect only a single rating per time stamp, i.e. $n^{(t)} = 1$. To capture anomalies at multiple consecutive points in time, the work of [7] has to use multiple anomalies, while in our work a single interval suffices.

At this point we want to mention the difference between outliers and anomalies [7]: While *outliers* are irregular behavior attributed to mostly random corruptions of the data (like, e.g., measurement errors), *anomalies* are irregular behavior that follow a specific pattern (like, e.g., time points with consistently low ratings due to a change in the product's quality). In our work, we consider anomalous behavior.

Giving the above information, the observed data at time t is modeled as a categorical mixture model defined by

$$x_i^{(t)} \sim \begin{cases} \text{Categorical}(\boldsymbol{\pi}^{(t)}) & \text{if } z_i^{(t)} = 0 \\ \text{Categorical}(\mathbf{o}_{k(t)}) & \text{if } z_i^{(t)} = 1 \end{cases} \quad (2)$$

$$z_i^{(t)} \sim \begin{cases} 0 & \text{if } k(t) = 0 \\ \text{Bernoulli}(r_{k(t)}) & \text{else} \end{cases} \quad (3)$$

Here, $z_i^{(t)}$ is the indicator variable showing which ratings are anomalies. $\boldsymbol{\pi}^{(t)} \in [0...1]^S$ is the vector describing the base behavior at time t, $\mathbf{o}_k \in [0...1]^S$ is the kth anomaly behavior, and r_k is the mixing proportion. The higher the value of r_k, the higher the proportion of anomalies within the corresponding interval. If no anomaly is present at time t, all variables $z_*^{(t)}$ are set to zero, corresponding of using only the base behavior at this point in time. Thus, the mixture model's components referring to the anomalies are constrained to specific intervals (cf. Figure 3).

For a Bayesian treatment, we equip the variables with corresponding prior distributions. We use

$$\mathbf{o}_k \sim Dir(\hat{\boldsymbol{\alpha}}) \qquad r_k \sim Beta(\hat{\alpha}, \hat{\beta})$$

due to the properties of conjugacy. The vector $\hat{\boldsymbol{\alpha}}$, for example, can be used to specify prior knowledge about potential anomalies (e.g. anomalies should represent primarily low ratings). In all of our experiments we use $\hat{\boldsymbol{\alpha}} = \mathbf{1}$ and $\hat{\alpha} = \hat{\beta} = 1$ corresponding to non-informative priors.

For the lower and upper bounds we exploit the idea that anomalies should appear primarily in short time intervals. That is, we assume that the probability to observe an anomaly over a very long time frame is lower than the probability to observe anomalies over only a few time points. We capture this effect by drawing the lower and upper bounds according to an exponential distribution controlled by the duration $\Delta^{(L_k,U_k)}$ of the anomaly interval. Formally

$$p(L_*, U_*) \propto \begin{cases} \prod_{k=1}^{K} e^{-\lambda \cdot \Delta^{(L_k,U_k)}} & \text{if disjoint intervals} \\ 0 & \text{else} \end{cases} \quad (4)$$

Please note that this is the joint distribution over all intervals since we require their disjointness. The larger λ, the larger the bias to small anomaly intervals. Per default, if no prior knowledge is given, one should select $\lambda = 0$. In this case, any combination of L_*, U_* is equally likely, corresponding to a non-informative prior. Note also that $\lambda = 0$ is a *valid* prior since the domain of L_*, U_* is finite.

Fig. 3: Illustration of the generative process using temporally constrained categorical mixture models

Part 2: Smoothness of the base behavior. So far, we have not specified any distribution on the base behavior $\boldsymbol{\pi}^{(t)}$. The core idea is that the base behavior should evolve smoothly over time according to the general behavior of the users. That is, we want to incorporate the temporal properties of the data.

As pointed out in [6], the (mean) parameters of the categorical distribution and their corresponding Dirichlet hyperparameters are not amenable to sequential modeling. Therefore, we exploit a similar idea as proposed in [6, 17]: instead of operating on the (mean) parameters $\boldsymbol{\pi}^{(t)}$, we operate on the natural parameters (cf. exponential family [5]). For the categorical distribution, the natural parameters are simply given by the logs of the mean values, i.e. $b_s^{(t)} = log(\pi_s^{(t)}/\pi_S^{(t)})$. While the mean parameters are restricted to lie on the simplex, the natural parameters are unconstrained, leading to an elegant way of sequential modeling.

In the following, we only operate on the natural parameters $\mathbf{b}^{(t)}$. If the mean parameters are required (e.g. as in Equation 2), we can simply apply the transformation

$$\pi_s^{(t)} = \frac{exp(b_s^{(t)})}{\sum_{j \in S} exp(b_j^{(t)})} =: \pi(\mathbf{b}^{(t)})_s$$

Note that the term $b_S^{(t)}$ is always 0. Therefore, we can ignore it for the remainder of the discussion, thus, operating effectively on an $S-1$ dimensional space.

Given the natural parameters $\mathbf{b}^{(t)}$ at each time index $t \in T$, we model their smoothness by exploiting the idea of linear state space systems [5] in combination with Brownian motion [11, 17]. First, we assume an underlying state space $\widetilde{\mathbf{b}}^{(t)}$ which temporally evolves over time via

$$\widetilde{\mathbf{b}}^{(t)} \sim \mathcal{N}(\widetilde{\mathbf{b}}^{(t-1)}, \Delta^{(t)} \cdot \mathbf{Q}) \quad (5)$$
$$\widetilde{\mathbf{b}}^{(1)} \sim \mathcal{N}(\widetilde{\mathbf{b}}_0, \mathbf{Q}_0)$$

We call this space the "smoothed" base behavior. Intuitively, the state of the smoothed base behavior at time t corresponds to the old state plus a small deviation government by the noise covariance matrix $\Delta^{(t)} \cdot \mathbf{Q}$. The larger the time difference between two observed ratings, the larger the corresponding covariance. That is, we effectively allow a higher change in the base behavior if the elapsed time between two ratings is high. If time points are very close to each other, we allow only small changes in the base behavior. In the limit, this discrete-time Gaussian random walk corresponds to Brownian motion [11, 17].

This process captures naturally the effects of rating data. In the case of movies, for example, one might see many rat-

ings appearing in short time frames during the time the movie has been released to the theaters and again many ratings a few month later when the DVD has been released. Both time frames potentially describe different base behavior due to different audiences.

Given the smoothed base behavior, the actual base behavior is now obtained by the simple random process

$$\mathbf{b}^{(t)} \sim \mathcal{N}(\widetilde{\mathbf{b}}^{(t)}, \mathbf{R}) \tag{6}$$

which again allows a small deviation between the base behavior and its smoothed counterpart. Note that we do not directly impose the temporal evolution between the variables $\mathbf{b}^{(*)}$, but via the state space $\widetilde{\mathbf{b}}^{(*)}$. This additional layer is in particular beneficial if the number of ratings varies strongly between the time points. If we would not use $\widetilde{\mathbf{b}}^{(*)}$, a single time point with a huge amount of ratings could dominate most of the temporal behavior.

Finally, we add corresponding priors to the newly introduced parameters. By exploiting the fact of conjugacy it follows that $\mathbf{Q}$ is drawn from an Inverse-Wishart distribution, i.e. $\mathbf{Q} \sim \mathcal{W}^{-1}(\Psi_q^0, \nu_q^0)$. The parameters Ψ_q^0 and ν_q^0 can be used to control the smoothness of the base behavior by providing prior knowledge about the noise covariance. Similarly, $\mathbf{R}$ follows an Inverse-Wishart distribution and $(\widetilde{\mathbf{b}}_0, \mathbf{Q}_0)$ a Normal-Inverse-Wishart distribution.

Summary and Discussion.

Overall, our generative process captures the temporal properties of the data by modeling a smooth base behavior as well as accounting for anomalies which are constrained to occur at certain time intervals. We will show in Section 3 how we perform efficient (approximate) inference for this model.

Model Selection. So far, we assumed the number K of anomalies is given. If not apriori known, we can estimate it via model selection. We choose the Bayesian information criterion [5]. Any other criterion can be used as well.

As we will see in Sec. 3, we will integrate out all latent variables except of $\Theta = \{L_*, U_*, \boldsymbol{Q}, \boldsymbol{R}\}$. Thus, increasing the value of K by one, increases the number of free parameters in our model by about 2 (the lower and upper bound of the new anomaly interval). This is a slight overestimate since the intervals need to be disjoint and, thus, they are not completely independently free variables. Given this result, we can choose the K minimizing the BIC equation

$$BIC(K) = -2 \cdot \ln \mathcal{L}_K + (2 \cdot K + c) \cdot \ln(\sum_t n^{(t)})$$

Here, the constant c denotes the parameters of the model which do not increase when increasing K. Since the value of c does not affect the optimal choice for K, we can simply set it to 0. The term $\mathcal{L}_K$ denotes the likelihood of the data when using K anomaly intervals. We can approximate it with the technique shown in Section 3.

Prediction. Since the base behavior evolves via a linear state space system, we are able to predict the behavior at future points in time. Combining Eq. 5 and 6, it follows

$$\mathbf{b}^{(T+1)} \sim \mathcal{N}(\widetilde{\mathbf{b}}^{(T)}, \mathbf{R} + \Delta^{(T+1)} \cdot \mathbf{Q}) \tag{7}$$

Thus, given estimates for $\widetilde{\mathbf{b}}^{(T)}$, $\mathbf{R}$, and $\mathbf{Q}$ (cf. Section 3), comparing the observed ratings at time $T+1$ against the predicted base behavior can be used as an indicator whether new anomalies have been occurred.

3. ALGORITHM

While the previous section focused on the model's generative process, we now present our learning technique. That is, *given* a set of observations X we aim at inferring the values of the hidden variables which best describe the observed data. There are multiple ways to formulate this objective. In this work, we treat the variables $\Theta = \{L_*, U_*, \boldsymbol{Q}, \boldsymbol{R}\}$ as parameters and we are interested in finding their maximum a posteriori estimate Θ_{MAP} as well as the posterior distribution $p(V|X, \Theta_{MAP})$ of the latent variables $V = \{\mathbf{o}_*, r_*, z_*^{(*)}, \mathbf{b}^{(*)}, \widetilde{\mathbf{b}}^{(*)}\}$ (which can then, e.g., be used to pick specific realizations of the latent variables).

3.1 Variational EM

Since exact inference in our model is intractable, we compute an approximation using variational expectation-maximization [5]. The idea is to approximate $p(V|X, \Theta)$ by a tractable family of parametrized distributions $q(V|\Omega)$. The parameters Ω are the free variational parameters. These parameters are optimized such that the best approximation between q and p is obtained. This corresponds to the expectation step of the variational EM method. Technically, we minimize the Kullback-Leibler divergence between q and p by optimizing over Ω. Using Jensen's inequality, minimizing the KL divergence is equivalent to maximizing the following lower bound on the log marginal likelihood [5]:

$$\mathcal{L}(X; \Theta, \Omega) = \mathbb{E}_q[\ln p(X, V, \Theta)] + H(q) \tag{8}$$

where $\mathbb{E}_q[.]$ denotes the expectation w.r.t. the q distribution and H the entropy.

Given an approximation of $p(V|X, \Theta)$ via $q(V|\Omega)$, we then determine updated parameter values for Θ by again maximizing Equation 8. This corresponds to the maximization step of the EM method.[1]

In short, the general processing scheme of our method is to alternatingly maximize $\mathcal{L}(X; \Theta, \Omega)$ w.r.t. Ω and Θ. *As we will later see, we actually interweave both steps by simultaneously optimizing parts of* Θ *and* Ω.

3.1.1 Variational distribution

In contrast to the frequently used mean field approximation, which assumes a fully factorized distribution, we use

$$p(V \mid X, \Theta) \approx q(V|\Omega) := \prod_k q_1(\mathbf{o}_k) \cdot \prod_k q_2(r_k) \cdot \prod_t \prod_i q_3(z_i^{(t)}) \cdot \prod_t q_4(\mathbf{b}^{(t)}) \cdot q_5(\widetilde{\mathbf{b}}^{(1)}) \cdot \prod_{t>1} q_5(\widetilde{\mathbf{b}}^{(t)} \mid \widetilde{\mathbf{b}}^{(t-1)})$$

We retain the sequential structure of the smoothed base behavior in q_5. Indeed, as described later, we determine q_5 via a Kalman filter where it follows that $q_5(\widetilde{\mathbf{b}}^{(t)}|\widetilde{\mathbf{b}}^{(t-1)})$ is a Normal distribution given by $\mathcal{N}(\widetilde{\mathbf{b}}^{(t)} \mid \widetilde{\boldsymbol{\mu}}_{t|T}, \mathbf{P}_{t|T})$. For the remaining variational distributions we use

$$q_1(\mathbf{o}_k) = Dir(\mathbf{o}_k|\boldsymbol{\alpha}_k) \qquad q_3(z_i^{(t)}) = Bernoulli(z_i^{(t)}|\phi_{t,i})$$
$$q_2(r_k) = Beta(r_k|\alpha_k, \beta_k) \quad q_4(\mathbf{b}^{(t)}) = \mathcal{N}(\mathbf{b}^{(t)}|\boldsymbol{\mu}^{(t)}, v^{(t)} \cdot I)$$

where $\Omega = \{\boldsymbol{\alpha}_*, \alpha_*, \beta_*, \phi_{*,*}, \boldsymbol{\mu}^{(*)}, v^{(*)}\}$ are the variational parameters to be optimized.

[1]The only actual difference between these steps is that Θ represents a point estimate of the random variables, while Ω represents the hyperparameters of a full distribution.

Note that the distributions $q_3(z_i^{(t)})$ and $q_3(z_{i'}^{(t)})$ are identical when $x_i^{(t)} = x_{i'}^{(t)}$, i.e. when both ratings have the same value. Thus, in practice we do not need to keep track of $n^{(t)}$ many different distributions at time t but it is sufficient to record S many distributions; one for each possible evaluation. We denote with ϕ_t^s the variational parameter of the distribtion q_3 for all ratings showing evaluation s at time t.

3.1.2 Optimization Procedures

As described above, our goal is to update the values of Ω and Θ by maximizing (or more generally increasing) the value of Equation 8. One crucial requirement of our technique was to ensure the efficiency of our method. In the following, we want to highlight the most important results.

3.2 Optimizing the Lower/Upper Bounds

A first naive solution to update the lower/upper bounds of the anomaly intervals would be to test any possible combination. Obviously, this solution is not efficient and requires time $O(T^2)$ already for a single anomaly. We provide a principle which is *linear* in the number of time stamps.

We start with the case of a single anomaly and uniform gaps between all time stamps, i.e. it holds K=1 and $\Delta^{(t)}$=1 for all t.

3.2.1 Simultaneous Optimization

Equation 3 shows the dependency between L_1/U_1 and z. Intuitively, the bounds act as a switch on the distribution of z: if z is outside of the interval, it is the trivial 0 distribution; if it is inside, it is a Bernoulli. Accordingly, assuming the posterior distribution for z (or its approximation q_3) is *given*, an optimization of L_1/U_1 is rather meaningless since one trivially has to capture all time points where the distribution is not the constant 0. Therefore, we propose to *simultaneously* optimize L_1/U_1 and q_3 to maximize Equation 8.

Observation: If we know that a time point t fulfills $t \in [L_k, U_k]$, the optimal distribution of $q_3(z_i^{(t)})$ can be computed independent from all other points in time. The optimal distribution is obtained by setting its variational parameter $\phi_{t,i}/\phi_t^s$ to the value as derived in Sec. 3.3. In particular, this value is independent of the actual values of L_k and U_k (knowing that $t \in [L_k, U_k]$). Based on this result we can also compute the entropy

$$h_{t,s} := H(q_3(z_i^{(t)})) = -\phi_t^s \ln \phi_t^s - (1-\phi_t^s)\ln(1-\phi_t^s)$$

for all $z_i^{(t)}$ fulfilling $x_i^{(t)}$=s. If $t \notin [L_k, U_k]$, we have $q_3(z_i^{(t)} = 0) = 1$ and we define the entropy $H(q_3)$ to be zero.

Using these results and the derivations of the appendix, as well as removing all terms which are independent of L_1, U_1 and q_3, we can reformulate Equation 8 to:

$$\begin{aligned}
&\ln p(L_*, U_*) + \sum_{t\in T}\sum_{i\in n^{(t)}} \mathbb{E}_q[\ln p(z_i^{(t)}|...) + \ln p(x_i^{(t)}|...)] + H(q_3(z_i^{(t)})) \\
&= -\lambda\cdot\Delta^{(L_1,U_1)} + \sum_{t\notin[L_1,U_1]}\sum_{s=1}^{S} n_s^{(t)}\cdot\mathbb{E}_q\left[\ln\pi(\mathbf{b}^{(t)})_s\right] + \sum_{t\in[L_1,U_1]}\Big(\\
&\quad \sum_{s=1}^{S}[n_s^{(t)}\cdot h_{t,s} + n_s^{(t)}\cdot\phi_t^s\cdot(\mathbb{E}_q[\ln r_{k(t)}] + \mathbb{E}_q[\ln o_{k(t),s}]) \\
&\qquad + n_s^{(t)}\cdot[1-\phi_t^s]\cdot(\mathbb{E}_q[\ln(1-r_{k(t)})] + \mathbb{E}_q[\ln\pi(\mathbf{b}^{(t)})_s])\Big) \\
&= \lambda + \sum_{t\in T}\sum_{s=1}^{S} n_s^{(t)}\cdot\mathbb{E}_q\left[\ln\pi(\mathbf{b}^{(t)})_s\right] + \sum_{t\in[L_1,U_1]} f_1(t) \qquad (9)
\end{aligned}$$

where we used the fact $\Delta^{(L_k,U_k)} = U_k - L_k$ and we defined

$$\begin{aligned}
f_k(t) := -\lambda + \sum_{s=1}^{S} n_s^{(t)}\cdot\phi_t^s\cdot[\mathbb{E}_q[\ln r_k] + \mathbb{E}_q[\ln o_{k,s}] - \mathbb{E}_q\left[\ln\pi(\mathbf{b}^{(t)})_s\right]] \\
+ n_s^{(t)}\cdot[1-\phi_t^s]\cdot\mathbb{E}_q[\ln(1-r_k)] + n_s^{(t)}\cdot h_{t,s}
\end{aligned}$$

Intuitively, the function $f_1(t)$ measures the "gain" in the log-likelihood when adding t to the anomaly interval.

The first two terms of Eq. 9 can be removed since they are constant w.r.t. the bounds and q_3 and thus do *not* affect the optimal solution. Accordingly, maximizing Eq. 8/9 w.r.t. L_1, U_1, and q_3 is equivalent to solving

$$(L_1^*, U_1^*) = \arg\max_{(L_1,U_1)} \sum_{t=L_1}^{U_1} f_1(t) \quad \text{with } 1 \le L_1 \le U_1 \le T$$

Since the function f_1 is independent of L_1/U_1, i.e. the terms $f_1(t)$ are constant within the current optimization step, we can record all values $f_1(t)$ in a finite array of length T. Thus, the above problem corresponds to the *Maximum Subarray Problem*. Using Kadane's algorithm [4], this problem can be solved in time $O(T)$.

3.2.2 Non-uniform gaps between timestamps

So far, we assumed $\Delta^{(t)} = 1$ for all $t \in T$. We now generalize the above result to handle varying values for $\Delta^{(t)}$.

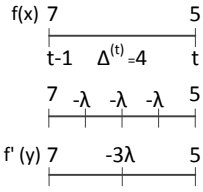

Fig. 4: Handling non-uniformity

W.l.o.g., due to finite precision in the measurement of time (e.g. UTC timestamps are usually measured in seconds), we can assume $\Delta^{(t)} \in \mathbb{N}^+$. Thus, a naive approach to handle the scenario of non-uniform gaps is to "blow up" the actual data by "artificial" time points where no ratings occur. After including the artificial time points, the Δ values are again equal to 1, and the previous technique can be applied. Figure 4 top and middle show this principle. Obviously, this principle is not suitable for huge time gaps and the new size of the array can be arbitrarily large.

Considering $f_1(t)$, it becomes apparent that its value evaluates to $-\lambda$ for each artificial time point. When searching for the subarray with maximal sum, these negative entries will never occur at the beginning/end of the anomaly interval [3]. If they would be at the beginning/end, one could easily shorten the interval to obtain a new one with higher sum. Thus, artificial time points are either completely contained in the interval or not included at all.

Using this result, we can safely "merge" all adjacent artificial time points to a single one with the function value $-\lambda\cdot u^{(t)}$, where $u^{(t)}$ is the number of artificial time points between time index t and $t-1$. Clearly, $u^{(t)} = \Delta^{(t)} - 1$ and the number of merged artificial time points is bounded by $T-1$. Overall, we can define a new array f' of size $2\cdot T - 1$ where

$$f_k'(y) = \begin{cases} f_k(\frac{y+1}{2}) & \text{if } y \text{ is odd} \\ -\lambda\cdot(\Delta^{(\frac{y}{2}+1)} - 1) & \text{if } y \text{ is even} \end{cases}$$

for $y \in [1, 2\cdot T - 1]$. And we now solve the problem

$$(a^*, b^*) = \arg\max_{(a,b)} \sum_{y=a}^{b} f_1'(y)$$

and set $(L_1^*, U_1^*) = (\frac{a^*+1}{2}, \frac{b^*+1}{2})$. Since the size of f' is bounded by $2\cdot T - 1$ the runtime complexity is $O(T)$.

3.2.3 Extension to multiple intervals

We now extend our result to multiple different anomalies/intervals. Using multiple anomalies affects the choice of the optimal q_3 distribution (cf. paragraph 'Observation' in Section 3.2.1). It is no longer sufficient to know whether $t \in [L_k, U_k]$ but we also have to know *which* k we consider.[2] Accordingly, for each anomaly k we have to use its individual function f_k / f'_k to measure the gain of adding a time point to the anomaly interval k. Overall, maximizing Eq. 8 using multiple intervals corresponds to solving

$$\arg\max_{(a_1,b_1),\ldots,(a_K,b_K)} \sum_{k\in K} \sum_{y=a_k}^{b_k} f'_k(y) \text{ with } a_k \le b_k < a_{k+1} \quad (10)$$

We solve the above problem by a dynamic programming technique. The necessary recursions are given by

$$g(1,1) = f'_1(1) \qquad g(1,t) = \max\{g(1,t-1) + f'_1(t), f'_1(t)\}$$
$$g(k,k) = m(k-1,k-1) + f'_k(k)$$
$$g(k,t) = \max\{g(k,t-1) + f'_k(t), m(k-1,t-1) + f'_k(t)\}$$
$$m(k,k) = g(k,k) \qquad m(k,t) = \max\{g(k,t), m(k,t-1)\}$$

Here, $m(k,t)$ (for $t \ge k$) denotes the maximal value of Eq. 10 when using k intervals and data up to point t. In contrast, $g(k,t)$ denotes the maximal value of Eq. 10 when the kth interval is forced to end at the t-th point in time (using k intervals and data up to t). Obviously, $g(k,t) \le m(k,t)$ holds. The value of $m(K,T)$ is the optimal value of Eq. 10.

We only provide a brief idea of these recursions: Assume we know the optimal intervals when using $k-1$ anomalies and data up to time point $t-1$. Let these intervals be denoted $(L_1,U_1),\ldots,(L_{k-1},U_{k-1})$. Additionally, assume the optimal intervals for k anomalies and data up to time point $t-1$ are given, denoted with $(L'_1,U'_1),\ldots,(L'_k,U'_k)$. Finally, assume the optimal intervals are given when the last interval is forced to end at time $t-1$ (we call these the g-intervals). Denote these with $(\hat{L}'_1,\hat{U}'_1),\ldots,(\hat{L}'_k,\hat{U}'_k)$, here $\hat{U}'_k = t-1$.

How will the solution for k intervals and data up to t look like? We can distinguish the following cases: (1) The time point t will be included in the optimal intervals. Obviously, since we are at the last point in time, it can only be represented by the kth interval. We can distinguish two subcases: (1a) The time point t is the beginning of the kth interval. In this case, the optimal intervals are $(L_1,U_1),\ldots,(L_{k-1},U_{k-1}),(t,t)$ and $m(k,t) = m(k-1,t-1) + f'_k(t)$. Since the last interval already ends at t, we also have $g(k,t) = m(k-1,t-1) + f'_k(t)$. (1b) The time point t is *not* the beginning of the kth interval. Thus, the optimal solution needs to be $(\hat{L}'_1,\hat{U}'_1),\ldots,(\hat{L}'_k,t)$ and we obtain $m(k,t) = g(k,t) = g(k,t-1) + f'_k(t)$.

(2) The time point t will *not* be included in the optimal intervals. In this case, since we want to find k intervals, the optimal solution is $(L'_1,U'_1),\ldots,(L'_k,U'_k)$ and $m(k,t) = m(k,t-1)$. For the g-intervals we have to distinguish two cases: (2a) The time point t is the beginning of the kth g-interval. In this case, the new g-intervals are $(L_1,U_1),\ldots,(L_{k-1},U_{k-1}),(t,t)$ and $g(k,t) = m(k-1,t-1) + f'_k(t)$. (Note that we use the optimal intervals from $m(k-1,t-1)$, not the g-intervals!) (2b) The time point t is *not* the beginning of the kth g-interval. Thus, leading to the solution $(\hat{L}'_1,\hat{U}'_1),\ldots,(\hat{L}'_k,t)$ with $g(k,t) = g(k,t-1) + f'_k(t)$.

[2]Technically, we could write $\phi^s_{t,k}$ to denote the optimal hyperparameter of q_3 assuming $t \in [L_k,U_k]$. We omitted k for brevity.

Exploiting the fact $g(x,y) \le m(x,y)$ and that we want to maximize $m(x,y)$, leads to the recursion as defined above. It is easy to add data structures to the method which record the start/end positions of the optimal intervals. Solving the above recursions via dynamic programming, we obtain:

THEOREM 1. *The optimal values for L_*, U_* and the optimal distributions q^*_3 can be computed in time $O(K \cdot T)$.*

3.3 Optimization of q_1, q_2, q_3

Following the principle of [5], the optimal distribution for q_3 can be determined by

$$\ln q^*_3(z_i^{(t)}) = \mathbb{E}_{q \setminus z_i^{(t)}}[\ln p(X,V,\Theta)] + const$$

Here, the constant *const* absorbs all terms which are independent of $z_i^{(t)}$ and, thus, do not affected the optimal distribution of q_3. The term $\mathbb{E}_{q \setminus z_i^{(t)}}[.]$ denotes the expectation with respect to the distribution q taken overall all variables except of $z_i^{(t)}$. Assuming $k(t) = k \neq 0$, and using the results from the appendix, it follows that

$$\ln q^*_3(z_i^{(t)} = 1) = \mathbb{E}_q[\ln r_k] + \mathbb{E}_q[\ln o_{k,s}] =: x$$

$$\ln q^*_3(z_i^{(t)} = 0) = \mathbb{E}_q[\ln(1-r_k)] + \mathbb{E}_q[\ln \pi(\mathbf{b}^{(t)})_s] =: y$$

where $s = x_i^t$. Therefore, the optimal value of the variational parameter is $\phi_{t,i} = \phi^s_t = \frac{e^x}{e^x + e^y}$.

The same principle can be applied for the distributions q_1 and q_2, leading to

$$\alpha_{k,s} = (\hat{\boldsymbol{\alpha}})_s + \sum_{t=L_k}^{U_k} n_s^{(t)} \cdot \phi^s_t$$

$$\alpha_k = \hat{\alpha} + \sum_{t=L_k}^{U_k} \sum_{s=1}^{S} n_s^{(t)} \cdot \phi^s_t \qquad \beta_k = \hat{\beta} + \sum_{t=L_k}^{U_k} \sum_{s=1}^{S} n_s^{(t)} \cdot (1-\phi^s_t)$$

3.4 Remaining Optimizations

Optimizing the base behavior. The base behavior can be updated for each $\mathbf{b}^{(t)}$ independently. Removing all terms from Equation 8 which are independent of $\mathbf{b}^{(t)}$ leads to

$$\mathbb{E}_q[\sum_i \ln p(x_i^{(t)}|...)] + \mathbb{E}_q[\ln p(\mathbf{b}^{(t)}|\widetilde{\mathbf{b}}^{(t-1)}, \mathbf{R})] + H(q_4(\mathbf{b}^{(t)})) \quad (11)$$

The first term is given in the appendix, and $H(q_4(\mathbf{b}^{(t)})) = \frac{S-1}{2}\ln(2\pi e v^{(t)})$. For the second term we derive:

$$\mathbb{E}_q\left[\ln p(\mathbf{b}^{(t)} \mid ...)\right] = -\frac{1}{2}\cdot\mathbb{E}_q\left[(\mathbf{b}^{(t)} - \widetilde{\mathbf{b}}^{(t)})^T \cdot \mathbf{R}^{-1} \cdot (\mathbf{b}^{(t)} - \widetilde{\mathbf{b}}^{(t)})\right] + c_1$$

$$= -\frac{1}{2}\mathbb{E}_q\left[\mathbf{b}^{(t)^T} \cdot \mathbf{R}^{-1} \cdot \mathbf{b}^{(t)}\right] + \mathbb{E}_q\left[\mathbf{b}^{(t)^T} \cdot \mathbf{R}^{-1} \cdot \widetilde{\mathbf{b}}^{(t)}\right] + c_2$$

$$= -\frac{1}{2}\left[Tr(\mathbf{R}^{-1} \cdot v^{(t)}) + \boldsymbol{\mu}^{(t)^T} \cdot \mathbf{R}^{-1} \cdot \boldsymbol{\mu}^{(t)}\right] + \widetilde{\boldsymbol{\mu}}^T_{t|T} \cdot \mathbf{R}^{-1} \boldsymbol{\mu}^{(t)} + c_3$$

We absorbed all terms which are independent of $\mathbf{b}^{(t)}$ into the constants c_i. Overall, Eq. 11 can be written as a function of $\boldsymbol{\mu}^{(t)}$ and $v^{(t)}$, which we optimize using gradient ascent.

Optimizing the smoothed base behavior. Since our model corresponds to a linear system, we can use a Kalman filter/smoother to determine the distribution of q_5. We use the Rauch-Tung-Striebel smoother. Thus, the distribution of q_5 can be computed efficiently by a forward and backward pass, leading to an update with runtime complexity $O(T)$.

Since the outputs $\mathbf{b}^{(t)}$ of the dynamic system are not observations but distributions, we slightly adapt the Kalman

update/innovation equations. Following the standard calculus of Kalman filters, the predicted mean and covariance matrix for time t (given data up to time $t-1$) are given by $\widetilde{\boldsymbol{\mu}}_{t|t-1} = \widetilde{\boldsymbol{\mu}}_{t-1|t-1}$ and $\mathbf{P}_{t|t-1} = \mathbf{P}_{t-1|t-1} + \Delta^{(t)} \cdot \mathbf{Q}$. Given the measurement at time t, the measurement residual can be computed as $\mathbf{e}_t = \mathbb{E}_q[\mathbf{b}^{(t)}] - \widetilde{\boldsymbol{\mu}}_{t|t-1}$. Accordingly, the residual covariance is given by $\mathbf{S}_t = \mathbf{P}_{t|t-1} + \mathbf{R} + v^{(t)} \cdot I$. Note the increased variance due to the uncertainty of the base behavior. Letting the Kalman gain be defined by $\mathbf{K}_t = \mathbf{P}_{t|t-1} \cdot \mathbf{S}_t^{-1}$, we see that the Kalman gain is smaller for time points showing a high variance, i.e. high uncertainty, in the base behavior. These points affect the smoothed base behavior less strongly.

Continuing with the standard calculus, the updated mean and covariance are $\widetilde{\boldsymbol{\mu}}_{t|t} = \widetilde{\boldsymbol{\mu}}_{t|t-1} + \mathbf{K}_t \cdot \mathbf{e}_t$ and $\mathbf{P}_{t|t} = (I - \mathbf{K}_t) \cdot \mathbf{P}_{t|t-1} \cdot (I - \mathbf{K_t})^T + \mathbf{K}_t \cdot \mathbf{R} \cdot \mathbf{K}_t^T$. Here, we used the Joseph form of the covariance update equation since it holds for any value of $\mathbf{K}_t$. For the backward pass, the RTS smoother leads to $\widetilde{\boldsymbol{\mu}}_{t|T} = \widetilde{\boldsymbol{\mu}}_{t|t} + C_t \cdot (\widetilde{\boldsymbol{\mu}}_{t+1|T} - \widetilde{\boldsymbol{\mu}}_{t+1|t})$ and $\mathbf{P}_{t|T} = \mathbf{P}_{t|t} + C_t \cdot (\mathbf{P}_{t+1|T} - \mathbf{P}_{t+1|t}) \cdot C_t^T$, with $C_t = \mathbf{P}_{t|t} \cdot \mathbf{P}_{t+1|t}^{-1}$. Due to space limitations, we kindly refer to the rich literature on Kalman filter/smoother, for details about the derivations.

Optimization of Q and R. Updating $\mathbf{Q}$ and $\mathbf{R}$ follows from the properties of the conjugate prior. Note that Eq. 5 can also be written in the form $\widetilde{\mathbf{b}}^{(t)} - \widetilde{\mathbf{b}}^{(t-1)} \sim \mathcal{N}(0, \Delta^{(t)} \cdot \mathbf{Q})$. Also, by the definition of the normal distribution it holds that $\mathcal{N}(\mathbf{x}|0, \Delta \cdot \Sigma) = \Delta^{-d/2} \cdot \mathcal{N}(\mathbf{x}/\sqrt{\Delta}|0, \Sigma)$, where d is the dimensionality of the distribution. Since the terms $\Delta^{-d/2}$ are constant when optimizing the log-likelihood, they can be ignored. Thus, $\mathbf{Q}$ can be seen as the covariance matrix of a Normal distribution with *known* mean of zero. Correspondingly, we can use the Inverse-Wishart distribution as its (conjugate) prior. Following the results of conjugacy, the posterior distribution of $\mathbf{Q}$ is an Inverse-Wishart distribution $\mathcal{W}^{-1}(\Psi_q, \nu_q)$ with $\nu_q = \nu_q^0 + T - 1$ and scale matrix

$$\Psi_q = \Psi_q^0 + \sum_{t=2}^{T} \mathbb{E}_q \left[\frac{1}{\Delta^{(t)}} \left(\widetilde{\mathbf{b}}^{(t)} - \widetilde{\mathbf{b}}^{(t-1)} \right)^T \cdot \left(\widetilde{\mathbf{b}}^{(t)} - \widetilde{\mathbf{b}}^{(t-1)} \right) \right]$$

which can easily be computed by plugging in the known expectations. Given this distribution, the MAP estimate for $\mathbf{Q}$ can efficiently determined by selecting the mode of the Inverse-Wishart distribution, i.e. $\mathbf{Q}^* = \frac{1}{(S-1)+1+\nu_q} \cdot \Psi_q$. The same principle can be applied for $\mathbf{R}$.

3.5 Overall Processing and Complexity

Using the above optimizations and update equations, our method iteratively recomputes the values for Θ and Ω. If the change in Equation 8 is less than 0.1% we assume convergence and terminate. Based on the previous results, and assuming that $K, S << T$, each iteration is linear in the number of time stamps, i.e. we have a complexity of $O(T)$.

4. RELATED WORK

Spotting anomalies in rating data: So far, only [7] considers the temporal analysis of rating data incorporating potentially anomalous behavior. The work models the rating data as distributions over time. As mentioned in the introduction, it requires an aggregation/binning of the data and it cannot handle intervals of anomalies. We compare our technique against [7] in the experimental analysis.

Modeling of temporal continuous data: Similar to the work [7], traditional time series modeling methods such as vector autoregression [14, 13] or Kalman filter/smoother [5], analyze continuous data. They are not directly suited for our scenario of categorical data (or require a problematic binning). Furthermore, traditional approaches for time series modeling are sensitive to outliers. Thus, these models fail to find good approximations of the data corrupted by anomalies. Therefore, robust techniques to handle outliers have been proposed [16]. These methods are designed to handle *outliers* which are attributed to mostly independent, random corruptions of the data, while our work is designed to handle *anomalies* following a specific pattern.

Since in our work the Kalman filter operates on the (clean) base behavior, i.e. the anomalies have been 'removed' by the other mixture model components, the problem of anomalies is circumvented. We compare our technique against a Kalman filter in the experimental analysis.

Modeling of temporal documents: One might represent the ratings at a certain point in time as a document with the words corresponding to the ratings' evaluations. Modeling temporal document collections is handled by dynamic topic mining [6, 17, 2]. Applying these methods on the 'documents' generated via the ratings is questionable since each document most likely would contain only a single 'word'. Ignoring this issue, further problems for our scenario are: First, [6, 2] require a binning of the documents in fixed time slots. Second, [6, 17] require that topics exist over the whole lifetime. In our work, however, anomalies exist only in specific time intervals. While [2] allows topics to appear and disappear, they prefer smooth evolutions. In our case, however, anomalies abruptly appear/disappear in time. Also, all of these techniques are (of course) designed to detect multiple topics. In our scenario, however, we want to find a single base behavior which captures the general temporal evolution, enriched by a few number of anomalies.

Related applications: Multiple techniques have been proposed in the area of *outlier detection* [1]. While the majority of techniques tackles the case of independently distributed data, time-series outlier detection and outlier detection for streaming data are also an active field of research [1]. Both areas differ from our work since they are designed for continuous data. Also, most existing techniques consider outlier in the sense of independent, random errors in the data. *Change detection* techniques detect points in time where the state of the underlying system has changed [15]. A change might not generally indicate anomalous behavior. Indeed, even the base behavior might change over time.

Studying product ratings has been done in multiple research areas, all following different goals and objectives. *Recommender Systems* incorporate ratings and their temporal information [9, 10] to improve the prediction performance. *Opinion mining* aims at extracting the sentiment of users regarding specific products or features of a product [18]. *Modeling* of social rating networks, e.g., to compactly describe the underlying mechanism driving the network or to generate synthetic data, has been studied, e.g, in [12].

None of the existing methods is designed to detect anomalies and the underlying evolving base behavior in rating data.

5. EXPERIMENTAL ANALYSIS

We applied our method (called SpotRate due to its potential to spot anomalies in rating data) on *over six million* product ratings representing varies categories: an extract of the Amazon website [8] evaluating multiple different products, another subset of the Amazon website evaluating

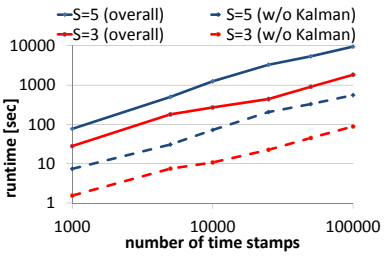

Fig. 5: Runtime vs. number of time stamps

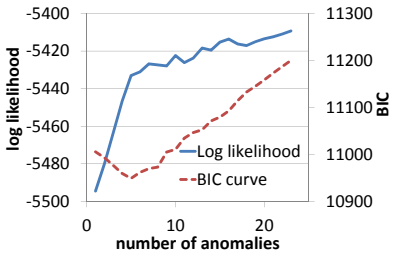

Fig. 6: Likelihood and BIC vs. K (synthetic data)

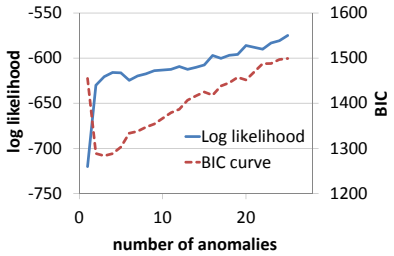

Fig. 7: Likelihood and BIC vs. K (real world data)

food products[3], ratings of restaurants in the area of Phoenix based on Yelp, and an extract of the TripAdvisor website[4] for hotel ratings. The data consists of tuples representing the ID of a product/service to be rated, the user who evaluated the product, the time stamp when the rating occurred, and a star rating in the range from 1 up to 5. Additionally, these datasets contain textual reviews, which we used to understand and describe the results of our method.

Besides these real world datasets we used synthetic data generated based on the presented process to analyze the scalability and robustness of our method.

5.1 Runtime Analysis

We briefly analyze the runtime of SpotRate. The runtime is primarily affected by the number of time stamps T and the rating scale S. The actual number of ratings does not affect the runtime (cf. Sec. 3.1.1). For the runtime analysis, we selected the product B00003TL7P from the Amazon dataset and we extended it to different length (from 1,000 to 100,000) by concatenation. Besides the original rating scale of $S = 5$, we used a rating scale of $S = 3$ by merging 1/2 and 4/5 ratings. All experiments were conducted on commodity hardware with 3 GHz CPU's and 4 GB main memory.

The results are shown in Fig. 5. Confirming our study of Sec. 3.5, the runtime increases linearly, showing the method's high scalability (note the slope of 1 in the log-log plot). The overall runtime for 100,000 time stamps (which would corresponds to 273 years when measured on a daily basis) is only about 158 minutes on commodity hardware. A brief study shows that the currently most rated products have around 20,000 (Amazon: Kindle Fire) or 8,000 (Yelp: Bottega Louie) ratings. Thus, even when considering the finest granularity, we highly exceed this number.

Additionally, we measured the runtime of our method when ignoring the time required for the Kalman smoother (dashed lines). As shown, the Kalman smoother contributs to around 90% of the absolute runtime. The remaining parts of our method are highly efficient.

We also studied the effect of the number of anomalies K on the runtime. According to Sec. 3.2.3, K linearly affects the runtime of the dynamic programming technique. Since the Kalman smoother (whose runtime is independent of K) accounts for most of the absolute runtime, we only observed a very small change of only a few seconds. Thus, overall, only T and S influence our method's practical applicability.

5.2 Effectiveness

In the following, we analyze the effectiveness of SpotRate considering different aspects. We start with the model selection principle. For this experiment, we generated synthetic data according to our model. We used 4000 ratings with 5 anomaly intervals. Figure 6 shows on the (first) y-axis the obtained log-likelihood of our method when varying the number K of potential anomalies. Obviously, the general trend shows that increasing K also increases the log-likelihood: more flexibility to describe the data is given. A very high increases is obtained until the value of 5, which corresponds to the true number of anomalies. After this point, the benefit of allowing further anomalies decreases.

This effect is well captured by the BIC score, which is shown on the second y-axis of the figure. The minimal BIC value is obtained for the value of 5. Thus, the model selection principle introduced before can be used as a good indicator how to select the number of anomalies.

The same behavior can be observed for real world data as, e.g., shown in Fig. 7. Here we plotted the log-likelihood and BIC score for a coconut-water sold on Amazon (cf. Sec. 5.4). Again, one sees a clear minimum of the BIC value, indicating that three anomaly intervals describe the data very well.

Next, we analyze our iterative optimization. In Fig. 8 we analyze how the log-likelihood increases when we increase the number of iterations until convergence. That is, on the x-axis we count how often the variables have been updated, while the y-axis shows the log-likelihood. We plotted the curves for different values of K, again for the product B00003TL7P. As expected, the first iterations lead to the highest improvement in the log-likelihood. Still, we see an improvement in the later iterations, showing the effectiveness of the optimization step. As also shown in the previous experiment, a higher value of K leads to a better likelihood. Additionally, for this product, we observed that a smaller number of intervals can lead to a lower number of required iterations. In general, however, the difference in the number of iterations was not as significant as shown for this product.

Finally, we analyze the effect of λ. Per default, a value of 0 can be selected to realize a non-informative prior. In Figure 9, we varied the value of λ between 1 and 0. We selected $K = 10$. As shown, for larger values of λ, shorter intervals are preferred. In particular, for $\lambda = 1$ the average interval length is close to the shortest possible length of 1. For $\lambda = 0$ larger intervals are captured. Note that $\lambda = 0$ does not mean that the whole set of time stamps is represented as an anomaly interval. Even in the case of $\lambda = 0$, we only report time intervals where the behavior is anomalous.

5.3 Comparison with related techniques

We compare SpotRate against the related technique RLA [7] and a Kalman smoother. Doing a fair comparison between these approaches is challenging since the data they analyze and goals they follow are different. In particular, the

[3] `http://snap.stanford.edu/data/`

[4] `http://sifaka.cs.uiuc.edu/~wang296/Data/`

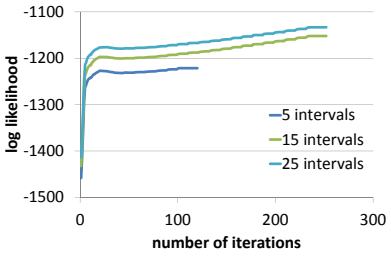

Fig. 8: Convergence analysis

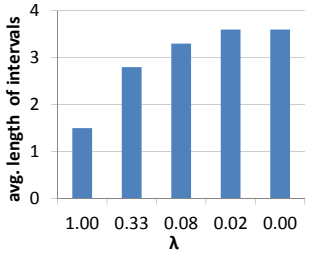

Fig. 9: Effect of λ

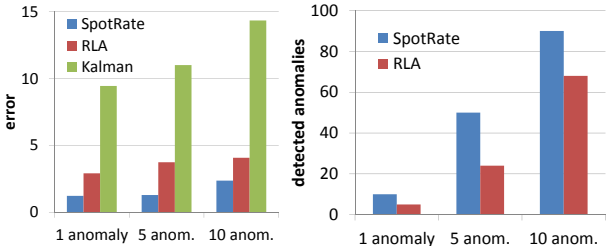

Fig. 10: Comparison with related methods

work [7] requires an aggregation of the data since it operates on the rating distributions. Thus, e.g., measuring the likelihood on the categorical data is questionable. Since RLA, however, is the only existing technique which also analyzes dynamic rating data, we try to study some effects.

For comparing the methods, we use two principles: In the first experiment, we compare the base behavior detected by the methods against the known base behavior for synthetically generated data. The base behavior is continuous in our model as well as in RLA, and for the Kalman smoother. Thus, it is fair to, e.g., measure the Frobenius norm between the ground truth base behavior and the detected ones. We generated data with 1000 time points and added a varying number of anomaly intervals to it, each covering 10 time points. We ensured that the anomaly intervals exactly match the aggregation required for RLA. Thus, this method gets a huge advantage since this assumption does not necessarily hold for real data. Fig. 10 (left) shows the results: Our method obtains the lowest error, it is able to detect the hidden base behavior. The Kalman smoother cannot handle anomalies and shows a high deviation to the ground truth.

In a second experiment, we evaluated whether the methods are able to detect the anomalous points in time (this is only possible for SpotRate and RLA). As shown in Figure 10 (right), our method almost perfectly detects all anomalous points in time. RLA in contrast is not able to spot all points, which also explains the previously observed higher error to the base behavior. Overall, our method outperforms the competing techniques in detecting the correct base behavior as well as spotting the anomalous points in time.

5.4 Discoveries

In the following, we will demonstrate the application of SpotRate by illustrating some of our interesting discoveries.

(1) We start with the example illustrated in Fig. 1. It represents a hotel in the Caribbean evaluated on TripAdvisor. While understanding the original time-stamped data is difficult, the extracted base behavior allows an easy understanding: clearly, the hotel is evaluated with mostly 4 and 5 stars. Our method found anomalous behavior in July and August 2005. In this time frame, the negative ratings highly increased. Analyzing the reviews at the detected time points, the reviewers criticized *"the restaurants with ridiculous reservation rules"* often showing overbooking and *"the nonfunctional air-conditions"*. These reviews indicate that in the given months the service of the hotel has dropped, potentially due to a highly increased number of guests. Our method was able to spot these anomalies, and it successfully smoothed out these points from the base behavior.

(2) Next, we show the result for a coconut-water sold on Amazon (http://www.amazon.com/dp/B000CNB4LE). Applying our method leads to the base behavior as shown in Figure 11. The three detected anomaly intervals appear at the end of 2010. As shown next to the figure, the detected anomalies are described by distributions $\mathbf{o}_k$ representing primarily low ratings. They clearly deviate to the base behavior. Inspecting the product's reviews during these times, most customers are not satisfied with the *"new plastic bottles"* the manufacturer has introduced, leading to a bad taste. Later time points do not show this anomalous behavior, indicating that the manufacturer has solved this problem (*"I can understand a lot of the initial bad reviews as I thought the new plastic bottle had a bad after taste. ... I can say that the taste is much improved ..."*).

(3) Next, we want to show the benefit of extracting an evolving base behavior. Figure 12 shows the base behavior of a baby bouncer (B00005QI1G) from the Amazon data. Looking at its evaluation, it is recognizable that the majority of reviewers evaluated this product with 5 stars. At the later time points, however, the number of low and medium ratings increases. Note that these intervals are not classified as anomalies but they represent the general evolution of the product. A closer look at the product's reviews at these time points explains that over time the customers were more and more unsatisfied by the product since it is *"nice to play but not long lasting"* and the *"battery simply does not last very long with the vibrating feature"*.

Discoveries via prediction. Finally, we want to show the potential of our method to detect anomalies via prediction. According to Eq. 7, we can predict the base behavior at future points in time. By comparing it against newly arriving ratings, anomalies can be spotted. We removed from all restaurants of the Yelp dataset the last 10 points in time. We applied our method on the remaining data. Figure 13 shows three restaurants whose predicted base behavior (left bar [a] of each diagram) highly deviates to the observed ratings (right bar [b]), thus, potentially indicating anomalies.

Inspecting the reviews of the first restaurant, we see comments like *"I've been eating at Stacy's for over a year so it pains me to kill them but the service [...] was pathetic. [...] I don't know if its a new employee or something going wrong but I'm probably not going back..."*. Thus, indicating that the service quality of the (previously very highly rated) restaurant has suddenly dropped.

For the second restaurant, we observed comments like *"All the prices have went up"* and *"The picture of the menu and prices is out dated"*, which again indicates a recent deviation to the previous behavior, potentially due to increased prices.

Finally, the reason for the abruptly appearing low ratings of the third restaurant seems to be caused by expanding/remodeling the old building. The old atmosphere of the restaurant seems not to be preserved and the larger capacity could not be handled by the service staff: *"the expanded building is nice [...] but something was lost. we didn't have*

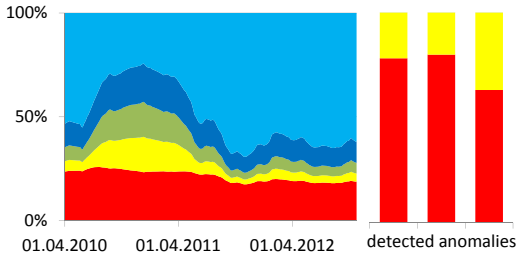

Fig. 11: Base behavior and anomalies (intervals spotted at the end of 2010).

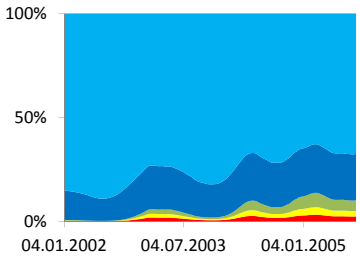

Fig. 12: Evolving base behavior (Amazon)

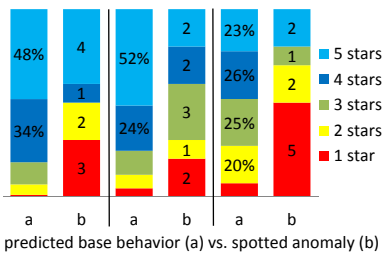

Fig. 13: Spotting anomalies via prediction (Yelp)

that hometown feel. we miss all the signatures and pictures." and *"I think its rad that you expanded, but if you cant handle the customer load then whats the point?"*.

Overall, by modeling the temporal evolution of the base behavior, our method is able to detect these newly occurring anomalies, which can then, e.g., be used to inform the corresponding companies.

6. CONCLUSION

We developed the method SpotRate for analyzing time stamped rating data. Our method detects the users' base behavior as well as time intervals representing anomalies. We proposed a sound Bayesian framework which represents the rating data via temporally constrained categorical mixture models. It accounts for the temporal evolution of the base behavior and enables us to predict the rating behavior for newly occurring ratings. We developed an efficient algorithm which exploits principles of variational inference and dynamic programming. Our experimental study has shown the potential of our method to spot anomalies and to use the base behavior for studying the evolution of a product.

Acknowledgments. Stephan Günnemann has been supported by a fellowship within the postdoc-program of the German Academic Exchange Service (DAAD). Research was also sponsored in part by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding parties. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

7. REFERENCES

[1] C. C. Aggarwal. *Outlier Analysis*. Springer, 2013.
[2] A. Ahmed and E. P. Xing. Timeline: A dynamic hierarchical dirichlet process model for recovering birth/death and evolution of topics in text stream. In *UAI*, pages 20–29, 2010.
[3] F. Bengtsson et al. Computing maximum-scoring segments in almost linear time. In *Computing and Combinatorics*, pages 255–264. Springer, 2006.
[4] J. Bentley. Programming pearls: algorithm design techniques. *Communications of the ACM*, 27(9):865–873, 1984.
[5] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2007.
[6] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *ICML*, pages 113–120, 2006.
[7] N. Günnemann, S. Günnemann, and C. Faloutsos. Robust multivariate autoregression for anomaly detection in dynamic product ratings. In *WWW*, pages 361–372, 2014.
[8] N. Jindal and B. Liu. Opinion spam and analysis. In *WSDM*, pages 219–230, 2008.
[9] N. Koenigstein, G. Dror, and Y. Koren. Yahoo! music recommendations: modeling music ratings with temporal dynamics and item taxonomy. In *RecSys*, pages 165–172, 2011.
[10] Y. Koren. Collaborative filtering with temporal dynamics. In *KDD*, pages 447–456, 2009.
[11] G. F. Lawler. *Introduction to stochastic processes*. CRC Press, 2006.
[12] K. Lerman. Dynamics of a collaborative rating system. In *WebKDD/SNA-KDD*, pages 77–96, 2007.
[13] R. B. Litterman. Forecasting with bayesian vector autoregressions. *Journal of Business & Economic Statistics*, 4(1):25–38, 1986.
[14] H. Lütkepohl. *New introduction to multiple time series analysis*. Cambridge University Press, 2005.
[15] X. Song, M. Wu, C. M. Jermaine, and S. Ranka. Statistical change detection for multi-dimensional data. In *KDD*, pages 667–676, 2007.
[16] J.-A. Ting, E. Theodorou, and S. Schaal. Learning an outlier-robust kalman filter. In *ECML*, pages 748–756, 2007.
[17] C. Wang, D. M. Blei, and D. Heckerman. Continuous time dynamic topic models. In *UAI*, pages 579–586, 2008.
[18] H. Wang, Y. Lu, and C. Zhai. Latent aspect rating analysis on review text data: a rating regression approach. In *KDD*, pages 783–792, 2010.

APPENDIX

Let $[\![.]\!]$ denote the Iverson bracket, it holds: $\mathbb{E}_q[[\![z_i^{(t)}{=}1]\!]]{=}\phi_{t,i}$
$\mathbb{E}_q[[\![z_i^{(t)}{=}0]\!]]{=}1-\phi_{t,i}$ $\quad \mathbb{E}_q[\ln \mathbf{o}_{k,s}]{=}\psi(\alpha_{k,s})-\psi(\sum_j \alpha_{k,j})$
$\mathbb{E}_q[\ln r_k]{=}\psi(\alpha_k){-}\psi(\alpha_k{+}\beta_k)$ $\mathbb{E}_q[\ln(1{-}r_k)]{=}\psi(\beta_k){-}\psi(\alpha_k{+}\beta_k)$

Given the definition of the distribution p, it follows:

- For $k(t)=k\neq 0$ it holds:
$\mathbb{E}_q[\ln p(z_i^{(t)}|\ldots)] = \mathbb{E}_q[\ln r_k^{[\![z_i^{(t)}=1]\!]}\cdot(1-r_k)^{[\![z_i^{(t)}=1]\!]}]$
$= \mathbb{E}_q[[\![z_i^{(t)}=1]\!]]\cdot\mathbb{E}_q[\ln r_k] + \mathbb{E}_q[[\![z_i^{(t)}=0]\!]]\cdot\mathbb{E}_q[\ln(1-r_k)]$
- $\mathbb{E}_q[\ln p(x_i^{(t)}|\ldots)] = \mathbb{E}_q[[\![z_i^{(t)}=1]\!]]\cdot\mathbb{E}_q[\ln o_{k(t),s}]+$
$\mathbb{E}_q[[\![z_i^{(t)}=0]\!]]\cdot\mathbb{E}_q[\ln\pi(\mathbf{b}^{(t)})_s]$ for $x_i^{(t)}=s$
- $\mathbb{E}_q[\sum_i \ln p(x_i^{(t)}|\ldots)] = \mathbb{E}_q\left[\sum_{s=1}^S n_s^{(t)}\cdot\ln p(x=s\mid\ldots)\right]$
$= \sum_{s=1}^S n_s^{(t)}\cdot\phi_t^s\cdot\mathbb{E}_q[\ln o_{k(t),s}] + n_s^{(t)}\cdot[1-\phi_t^s]\cdot\mathbb{E}_q[\ln\pi(\mathbf{b}_t)_s]$
- $\mathbb{E}_q\left[\ln\pi(\mathbf{b}^{(t)})_s\right] = \mathbb{E}_q\left[\ln\frac{e^{b_s^{(t)}}}{1+\sum_{s=1}^{S-1}e^{b_s^{(t)}}}\right] = \mathbb{E}_q\left[\mathbf{b}_s^{(t)}\right] -$
$\mathbb{E}_q\left[\ln(1+\sum_{s=1}^{S-1}e^{\mathbf{b}_i^{(t)}})\right] \geqslant \mu_s^{(t)} - \ln(\mathbb{E}_q\left[1+\sum_{s=1}^{S-1}e^{\mathbf{b}_s^{(t)}}\right])$
$= \mu_s^{(t)} - \ln(1+\sum_{s=1}^{S-1}e^{\mu_s^{(t)}+\frac{v^{(t)}}{2}})$