Application of Kalman Filters to Identify Unexpected Change in Blogs

Paul Logasa Bogen II, Joshua Johnston, Unmil P. Karadkar,
Richard Furuta, Frank Shipman
Center for the Study of Digital Libraries
and Department of Computer Science
Texas A&M University
College Station, TX 77843-3112 USA
walden@csdl.tamu.edu

ABSTRACT
Information on the Internet, especially blog content, changes rapidly. Users of information collections, such as the blogs hosted by technorati.com, have little, if any, control over the content or frequency of these changes. However, it is important for users to be able to monitor content for deviations in the expected pattern of change. If a user is interested in political blogs and a blog switches subjects to a literary review blog, the user would want to know of this change in behavior. Since pages may change too frequently for manual inspection for "unwanted" changes, an automated approach is wanted. In this paper, we explore methods for indentifying unexpected change by using Kalman filters to model blog behavior over time. Using this model, we examine the history of several blogs and determine methods for flagging the significance of a blog's change from one time step to the next. We are able to predict large deviations in blog content, and allow user-defined sensitivity parameters to tune a statistical threshold of significance for deviation from expectation.

ACM Categories and Subject Descriptors
H.3.7 [**Digital Libraries**]: *collection, systems issues, user issues.* H.3.3 [**Information Search and Retrieval**]: *Information filtering.* H.5.4 [**Hypertext/Hypermedia**]: *User issues.*

General Terms
Design, Human Factors, Experimentation, Management.

Keywords
Distributed Collection Management, Kalman Filters.

1. INTRODUCTION
Distributed digital collections are a part of several popular sites that comprise the community-centric banner of Web 2.0, such as digg.com, fark.com, and technorati.com. These collections differ from traditional digital libraries in that the maintainer of the distributed digital collection does not control the content of the members in their collection. In particular, a user of technorati.com faces challenges unique to its collection of blogs in that, unlike other collections, the content undergoes frequent and sometimes dramatic changes. While tasks such as notifying the user of any change are simple (through the use of technologies such as RSS), more complicated actions, where notifications are based on factors other than a simple detection of change, are not currently available.

In the past, we have examined change in blogs manually and concluded that blogs have regular patterns of change [5]. With a model of expectation of regular change, we hope to be able to detect unexpected changes.

In this paper, we describe a method for modeling expected blog content based on past content using Kalman filters. Kalman filters, like hidden Markov models, are powerful tools used to track the changing states of linear dynamical systems. This method lends itself to the identification of unexpected change by calculating the discrepancy between a predicted future state and the actual future state.

The rest of this paper is organized into six sections. We first discuss the problem of change in blogs. Second, we survey prior work regarding change on the Web, blog analysis, and Kalman filters. Third, we describe our methodology from data collection through the analysis of change. Fourth, we present results and a discussion of the effectiveness of Kalman filters in this context. This is followed by our planned future work, including a user study currently being developed. Finally, we present our conclusions on the use of Kalman filters for analyzing change given this initial exploration.

2. CHANGE IN BLOGS
Identifying significant change is often not a simple matter of setting a constant threshold on difference between two versions. Rather, it is a function of the past behavior of a document. Also, the significance of a change may not depend on if too much information has changed, but may depend on if too little information has changed. For example, if www.cnn.com stopped posting new articles, this would signify a significant change of behavior that would warrant a user's attention.

Since there exist cases where change can be desired and stability unwanted, what we seek to determine is not how much change is significant, but what pattern of change is expected. By modeling the expected amount of change, we can find unexpected deviations by calculating the difference between our expectation and the actual behavior.

JCDL'08, June 16–20, 2008, Pittsburgh, Pennsylvania, USA.

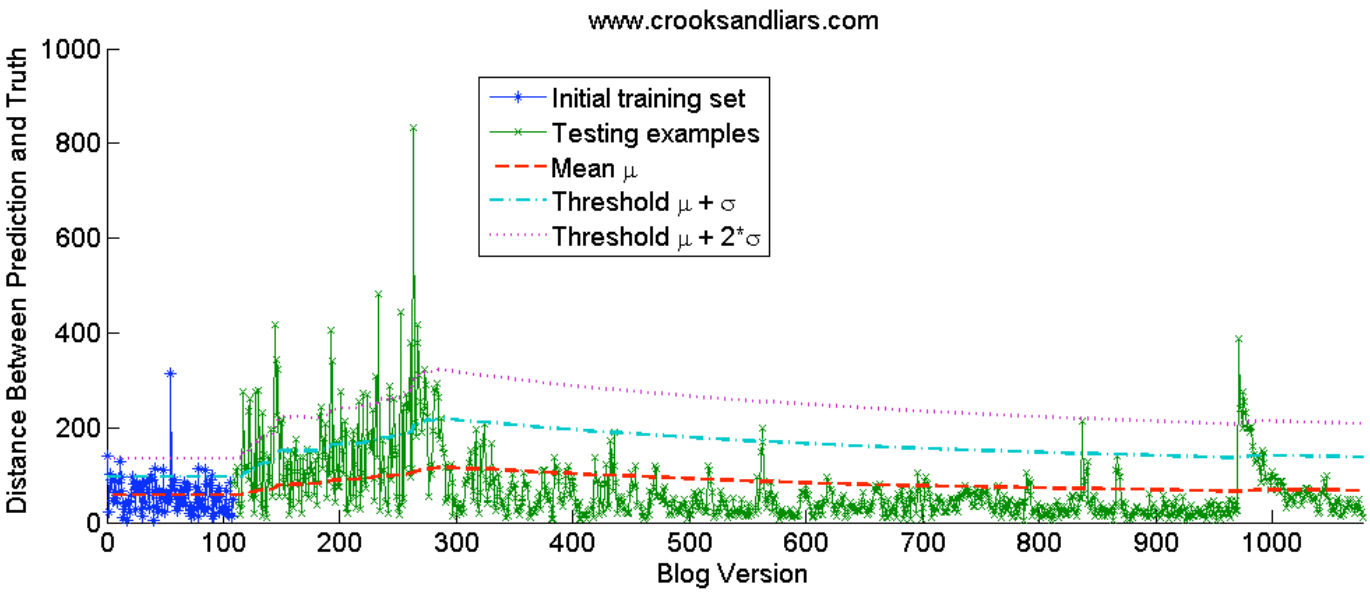

Figure 1 Kalman filtering results for the blog `www.crooksandliars.com`

While initial steps have been taken in the past [5], to the best of our knowledge the exploration of methods to model change has not been done. Therefore, to advance this line of inquiry, we are performing a preliminary test of Kalman filters as a potential method to model expected change. This technique allows us to identify unexpected change and provide a measure of its significance, allowing collection managers can track and manage blog collections based on change.

3. RELATED WORK

Related work in this area can be divided into four major areas: change-management of Web documents, blog analysis, our prior work on the Distributed Collection Manager, and Kalman filters.

3.1 Change and the Web

Since the early days of the Web, a large amount of research has been done in change-management methods. The Do-I-Care Agent treated changed documents as if a new document had been added to the collection for notification purposes [15]. Ashman described classes of document addressing strategies that assumed change was a problem to be resolved [1].

Additional research was also performed by Koehler on the rate of Web page change [11], Boese and Howe on how Web page changes can affect document genres [4], and Ivory and Megraw on the evolution of Web sites as a whole [9]. In each of these cases, the authors try to remove the damage that change inflicts on the relevance of collections. While Askehave and Nielsen recognize change as an integral part of Web documents, they place the analysis of such change outside the scope of their work and treat pages as static [2].

3.2 Analyses of Blogs

Prior works that analyze blogs have not been interested in how blogs change, but consist largely of comparisons between and classifications of blogs. Some have grouped blogs into genres [4], while others have grouped blogs based on social communities [6] or on spatio-temporal references [8].

3.3 Distributed Collection Manager

Our prior work on the Distributed Collection Manager has shown preliminary results that indicate different blogs may exhibit some common temporal behaviors [5]. This prior work inspected a subset of our data. This subset represents ten weeks of daily caches from September 25 to December 10, 2006. Our preliminary inspection consisted of measuring change between two versions of a Web resource with a standard vector-space cosine similarity metric. This change was measured in terms of angle. 90 degrees indicates a completely dissimilar page (orthogonal vectors), a page where there are no common terms between versions. 0 degrees indicates a page with no detected change (collinear vectors).

This preliminary inspection revealed two different patterns of behavior. First, we found that the change of blogs approached an asymptote with regards to our selected metric – as time progressed the absolute change compared to the initial cached version approached a "stable state" of about 62 degrees of change. Second, we discovered that blogs follow a weekly activity cycle. The relative change, change between two consecutive caches, displayed a recurring pattern where the magnitude of change declined on the weekends.

Our preliminary results confirmed our hypothesis that patterns of change exist for some classes of Web resources. For such resources, the significance of change to a resource in a collection may be highly correlated with its variance from the collection manager's expectations for change to that resource. This reinforced our belief that methods to predict change and model the characteristics of this change are needed in order to assess the impact of change, or lack thereof, on a collection.

3.4 Kalman Filters

Kalman filters are a successful tool for tracking the state of a linear dynamical system. They allow for not only the learning of the parameters for Equations 1 and 2, but also for the prediction of future values in a straightforward manner. We choose to use them

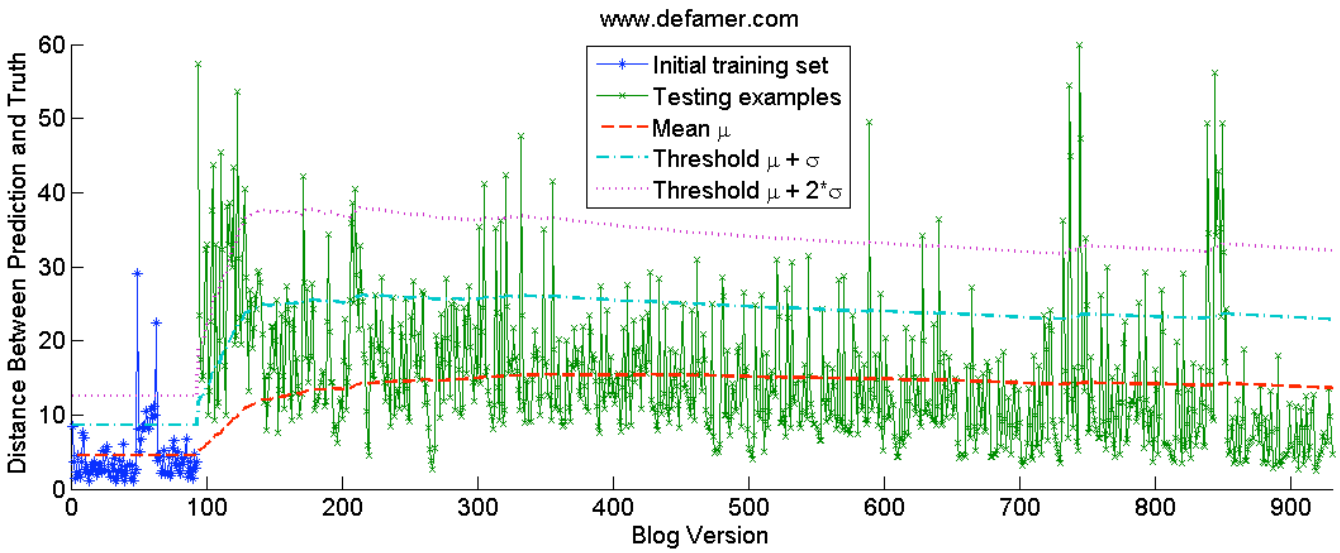

Figure 2 Kalman filtering results for the blog `www.defamer.com`

not only for their proven power [12][16], but also for their simplicity and direct approach.

Kalman filters are used to model systems that change over time, allowing predictions to be made about future states [16]. The underlying assumption in these dynamic systems is that future values depend on past values in a linear fashion. Observations, which may be noisy, can be made on the system, with some information remaining hidden (imperfect information). The state of the system is denoted with the variable x, which may have many hidden states. The state at a given time step t is denoted x_t. Observations about the system are denoted with the variable y_t. x_t and y_t are controlled by the following system of equations:

$$x_t = Ax_{t-1} + v_t, v_t \sim \acute{I}\ (0, Q) \quad (1)$$

$$y_t = Cx_t + w_t, w_t \sim \mathrm{N}(0, R) \quad (2)$$

A is a mixing matrix that determines how the system changes from time step $t-1$ to t. v is Gaussian noise introduced into the system at each time step, distributed with a mean at 0 and a covariance matrix Q. C is a mixing matrix that determines how we are allowed to view the observables in the system. Our observations also include Gaussian noise, w, distributed with mean at 0 and covariance R.

The canonical example for describing the use of a Kalman filter is in the tracking of an object moving through space. The state vector x might contain the current position of the object, as well as the velocities along each dimension. In 2D, x might look like $\left[x_1, x_2, dx_1, dx_2\right]^T$. The mixing matrix A would be formed such that at each time step, the position in each dimension is updated by the velocity. Say we are able to observe the position of the object in a 2D space (with noisy measurements), but not the current velocity of the object. Then, the matrix C would filter out the velocity values and give us a vector y containing $\left[x_1, x_2\right]^T$.

Recently, Kalman filters have begun to be used outside of their traditional application to sensor data modeling and are being applied to textual analyses. In particular, the area of topic tracking has seen great success using Kalman filters. Simultaneously, work has been conducted by Krause, et al., on topic intensity [10], Wang and McCallum on topic trends [18], Cselle, et al., on topic tracking in emails [7], and Blei and Lafferty on topic tracking in *Science* [3]. Additionally, Van Durme, et al., have used Kalman filters for semantic parsing for question answering systems [17].

With these successes in other text analysis domains, especially in a related field to change tracking such as topic tracking, the case for using Kalman filters to model changes to blogs gains strength. In the case of modeling blog change, we form the problem in such a way to make it analogous to a point moving in a high-dimensional space. Although it's not clear that blog content changes occur in a linear fashion, simple linear models often perform well. Therefore, we can use Kalman filters to estimate how the blog will change with time. In order to find unexpected change, we train a Kalman filter with the current history of a blog and use the filter to predict the next expected version. We can use a measure of distance from this expected version to the observed, true next version to tell us how unexpected the change is according to our model.

4. METHODOLOGY

This section describes the methodology used to evaluate the potential of Kalman filters to identify significant changes to blogs. This includes a description of the procurement and preprocessing of the blog data as well as the approach employed to track blog changes.

4.1 Data Collection

Over a period of one year, from March 2006 to March 2007, we cached a collection of 77 blogs. These blogs were obtained from `technorati.com`'s 100 most linked-to blogs, with 23 non-

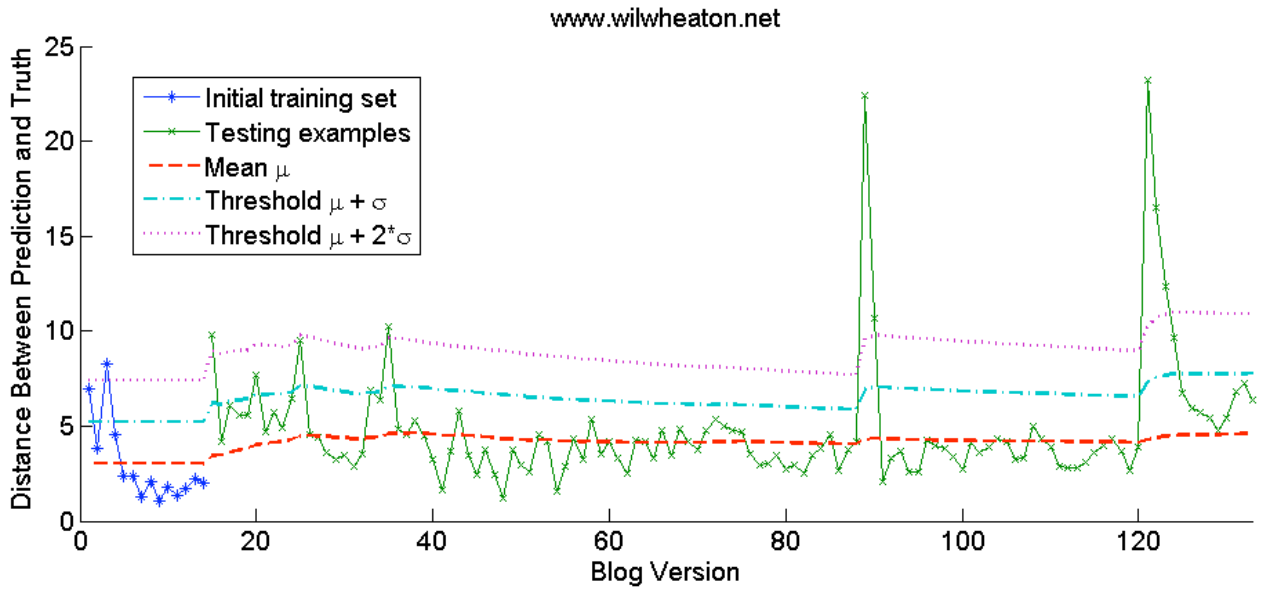

Figure 3 Kalman filtering results for the blog `www.wilwheaton.net`

English language blogs excluded. We selected a four-times-a-day caching period to allow us to perform day-to-day analysis with the additional redundancy of having four versions in a given day in case of a network or system outage. Our current work with Kalman filters uses all four daily caches. Each entry was stripped of HTML tags and JavaScript code. Then, using the Mallet library for Java [13], we removed stopwords and applied a custom stemming filter derived from the Porter2 English stemming algorithm [14]. Finally, Mallet was instructed to create dictionary-based term vectors which were loaded into MATLAB for analysis. We examined all the cached versions of each blog and removed adjacent identical entries. The rationale is that we're interested in the dynamics of change, when it occurs. If a user wishes to track stagnation, she can just monitor the amount of time since the last page update.

4.2 Preprocessing

Each blog contains a large vocabulary, usually on the order of 10,000 to 20,000 words (the dimension of the term vectors). This is too large a number to use for estimating parameters for a Kalman filter, which requires an estimate of the covariance of the terms. Therefore, we need to reduce the dimensionality of our data. Since we are primarily interested in analyzing the variance of our data, we select principal components analysis (PCA) as our dimensionality reduction technique. PCA is a common technique in pattern recognition to deal with cases where the large number of features collected makes the detection of meaningful patterns difficult. To overcome this problem, PCA attempts to find a simpler model of the data such that the overall variance is maximized to highlight different attributes. These methods enable us to effectively identify the more meaningful portions of our data set and focus on them, thus reducing the size of the problem. We use the "snapshot" PCA approach to avoid the computation of the full covariance matrix of the term vectors. After some empirical observation of performance with various numbers of principle components (including cases where using too many dimensions in our data caused memory errors in the MATLAB Kalman implementation we use), we chose to limit the number of components to the 25 highest eigenvalues (the 25 vectors of projection that account for the most variance in the data). Fewer than 25 components were used in cases where 90% of the variance in a blog could be accounted for with fewer components. The number of projected dimensions (chosen eigenvectors) can be adjusted as Kalman implementation and processor limitations warrant. The term vectors were then projected onto the reduced vector space. These projections were used to train and test our Kalman filters.

4.3 Modeling and Prediction

Since we treat blogs as points in a high dimension space (the number of eigenvectors, used for PCA, denoted p), our model consists of p observables, which are the values for each dimension at each time step. There are also p hidden variables, consisting of "velocities" along each dimension, representing the amount of expected change along each principal component from one blog to the next. Relating to Equations 1 and 2, the state of our system is $x_t = \left[x_1,\ldots,x_p,dx_1,\ldots,dx_p\right]^T$.

We start with 10% of the blog caches for training data. This represents a small snapshot of history to be used for future predictions. In our case it is safe to assume that without an idea of what is considered normal for a blog, we would have no reasonable way to estimate what is significantly different in future versions.

We train a Kalman filter on the set of training data and use the filter to make a prediction about the blog's contents at the next time step. We repeat this process by folding the test sample into the training set and retraining the Kalman filter, thus simulating the arrival of new data over time. Retraining the Kalman filter at each time step allows the model to adapt to gradual page drift and adapt to new content.

To gauge the accuracy of our predicted next version to the true next version of the blog, we use Euclidean distance between our prediction and the true next version (in the projected space). This gave us a sense of how far our prediction deviates from the truth.

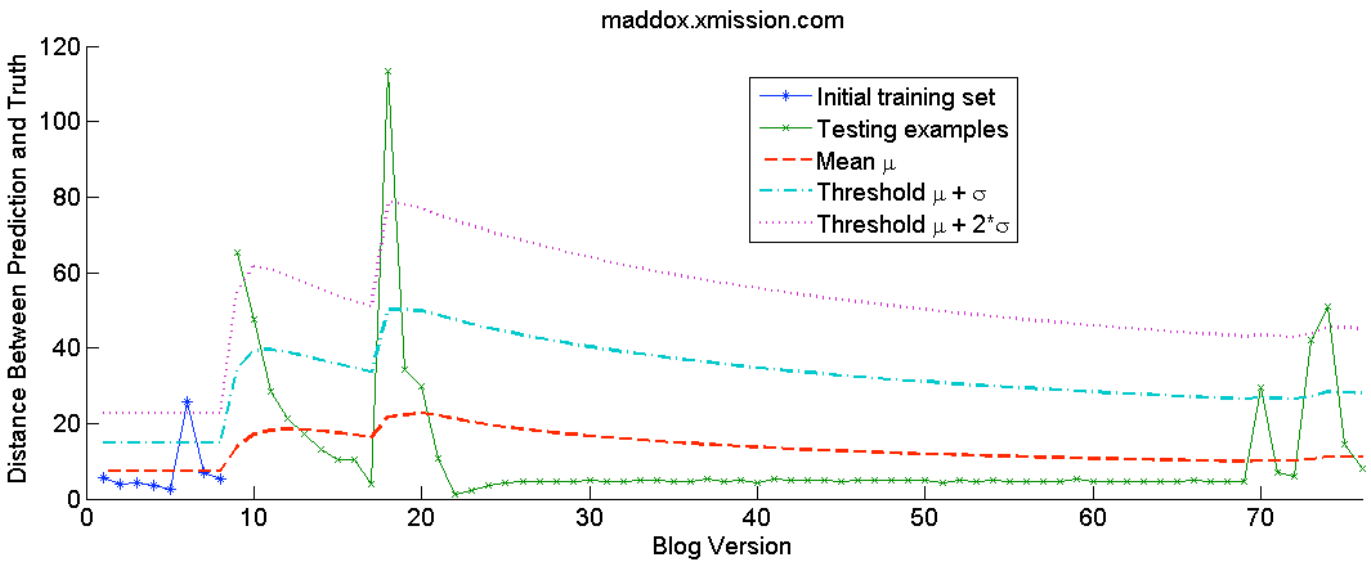

Figure 4 Kalman filtering results for the blog `maddox.xmission.com`

Blog changes with a large distance from the prediction are considered significant because the Kalman filter was unable to account for the change given the history.

Determining thresholds for significance can be difficult. Instead of using a constant distance threshold, which may be problematic since individual blogs might have different distances between samples for even insignificant changes, as well as distance ranges that may change over time, we use a threshold determined by the mean and standard deviation of prior distances. Starting with the training set, we compute the mean and standard deviation of the distances of the Kalman predictions to the true blog versions. At each time step, when we fold the new blog example into the training set, we re-compute the mean and standard deviation for the prediction distances. This gives us a moving average and a sense of how the variance of our predictions changes over time. The assumption is that in quiet periods, where predictions are easy to make, the mean and standard deviation will gradually settle to smaller values since our predictions are closer to the truth and have less prediction error. This means our model will become more sensitive, over time, to changes in stable blogs. Conversely, if a blog changes rapidly and our predictions are constantly far from the truth because of the blog's dynamic nature, the mean and standard deviation of the distances become large. This, in effect, made the filter less sensitive to change, so that fewer of the spikes in prediction error are flagged as significant.

Jumps in distance from predictions to the true blog versions are flagged as significant if

$$d \geq \mu + a\sigma \qquad (3).$$

Here, d is the distance from the prediction to the truth, μ and σ are the mean and standard deviations, respectively, of prior distances, and *a* is a user controlled parameter that allows a user to adjust the sensitivity of the filter. The rationale behind the use of this parameter is that the significance of changes is a subjective measure, and users may wish to adjust this value to suit their own needs. Higher values of *a* make the filter less sensitive to larger distances, while lower values of *a* mean the filter must be more accurate or a change is flagged as significant.

5. RESULTS AND DISCUSSION

Kalman filters provide a good estimate of significance in blog changes. We filter blogs using the methods described above and examine changes marked as significant by looking for distances that are either one or two standard deviations above the mean, $a = 1$ and 2 from Equation 3. Figures 1, 2, and 3 show the results of Kalman filtering on the blogs `www.crooksandliars.com`, `www.defamers.com`, and `www.wilwheaton.net`. The x-axis for each of the figures denotes time, where each point on the lines is one version of the blog. The y-axis measures the Euclidean distance of the Kalman predicted version of the blog's contents to the actual bag-of-words for that version. There are three smoother lines per blog, the bottom represents the moving average of the distances, and the upper lines represent the mean plus one and two standard deviations. Any blog version (marked with an 'x') that occurs above these thresholds might be considered to have a significant amount of change, depending on the user's value for *a* (Equation 3).

In Figure 2, Kalman filtering on `www.defamer.com` predicts a large amount of thrashing within the blog. Figure 1 shows a good deal of stability for `www.crooksandliars.com`, with relatively few spikes in significant deviations from the Kalman filter's predictions. The Kalman filtered data is less stable at first while it learns the blog's contents, but it settles over time such that at even one standard deviation, changes are rarely flagged as significant. As evident from the figures, the choice of *a* is important in determining the significance of blog changes and varies per blog. For noisy blogs, one may wish to set *a* to a high value in order to prevent too many notifications. For relatively stable blogs, lower values give the filter more sensitivity to flag changes. For `www.wilwheaton.net`, in the beginning, we see some fluctuation in the page contents during periods of where the blog was suffering technical difficulties, and then mostly static content until a few pages at the end where redirection pages are

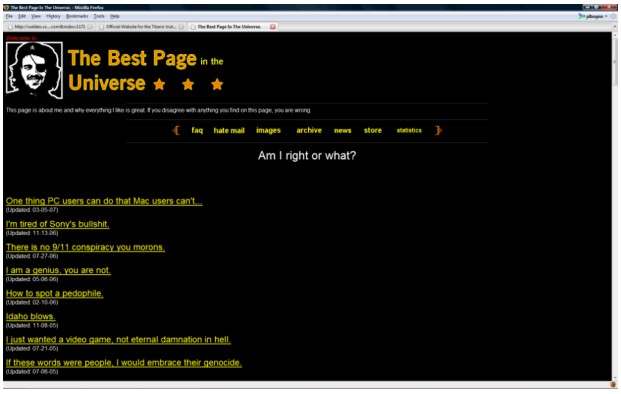

Figure 5 Expected `maddox.xmission.com` behavior.

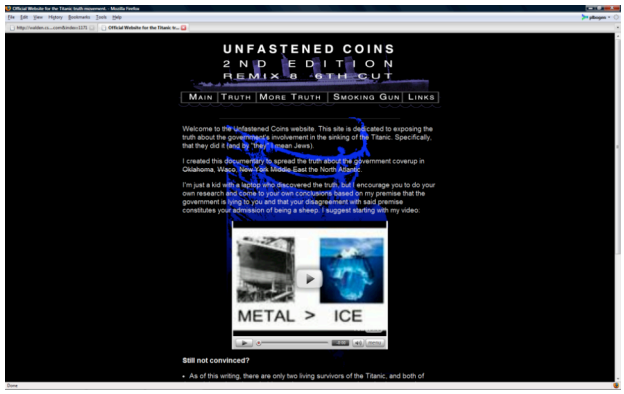

Figure 6 Unexpected `maddox.xmission.com` behavior.

put up and changed. There are only about 140 unique versions of the blog as the site becomes static and ceases to change.

We looked at the types of changes flagged as significant by Kalman filters. These versions were cases where the vocabulary content of a page, or the frequency of words, changed dramatically. This makes sense because the Kalman filters are tracking the term vector representations of blog content. If the vocabulary changes drastically, a large amount of difference from one term vector to another occurs. Many times, change was flagged as significant when many posts were made to a blog very quickly, completely replacing most of the text that was there before. When blogs changed gradually, with just a little text changing at a time, the filter was able to adapt to the slow page drift and match predictions well.

One more dramatic case we discovered was that of `maddox.xmission.com`. As can be seen in Figure 4, the errors of the prediction for all but five versions of the page fell within a standard deviation from the average amount of error. In each of the five cases where the prediction was off by greater than one standard deviation of the average error, abnormal behavior can be seen when the caches are manually inspected. In every other cached instance abnormal behavior did not occur. The first three spikes can be traced to temporary connection issues that resulted in a single corrupted cache. The last two spikes show a truly abnormal change on the site. The first of the two represents the sudden change in the page from being a collection of links to rants by the author, Figure 5, to a conspiracy theory page centered on the sinking of the Titanic, Figure 6. The last spike represents a reversion of the now expected conspiracy theory page back to the prior behavior.

Similar behaviors, albeit less dramatic, are found on every page we manually inspected against the measures of error in site prediction. Our comparison reveals that an error greater than two standard deviations from the average represents large changes in every instance. Some examples we saw were a blog that lost all of its articles, spikes in activity that caused a large number of articles to be posted or dropped off the front page, problems with the cache (either ours or theirs), and activity that is uncharacteristic of the blog, like `maddox.xmission.com`. While the empirical results are very promising, they do not provide the surety of a quantitative comparison between human and machine performance. To address this we are planning an evaluation, as described in the next section.

6. FUTURE WORK

In order to validate these and future models for blog change, we need to separate expected changes, such as a blog adding a new on-topic entry, from unexpected changes, such as a blog devoted to cars becoming a blog devoted to floral arrangement. In order to facilitate this, a study is currently being planned that has participants tag versions of blogs as being examples of either expected or unexpected change.

Besides the proposed evaluation, we would like to explore different methods for dimensionality reduction, including feature subset selection using standard information theoretic measures. Additionally, our implementation of Kalman filters forced us to set an upper bound on the number of principal components we could use to avoid memory errors. We'd like to find a way to work around this problem, or devise some adaptive maximum that isn't a hard threshold but can vary with different blogs. Also, there are several extensions to Kalman filters that we would like to explore. One example of these revisions include the use a sliding window of past history rather than all previous blog revisions when training the filter. This would give us better short term prediction accuracy and may help signify smaller deviations after a term of relative stability. The size of the window could be user dependent to control how much past information the user wants to be considered. We would also like to explore using particle filters as the system model, as they allow for non-linear transitions through time and non-Gaussian noise. It is not clear if blog changes undergo non-linear changes, as the Kalman filter seems to fit the data well in most cases. However, a particle filter might give more accurate predictions as it is a more flexible, albeit more complex, model.

The way we are using Kalman filters represents a regression technique, trying to estimate future values given prior behavior, or known values. Alternative regression techniques that are simpler than Kalman filters might also yield good results. Lastly, our approach may benefit by the addition of other heuristic-based features like document structure information, user provided input that hints at the user's expectation of change, or possibly other features used by NLP and stylometric categorizations, like punctuation usage and word sense disambiguation.

7. CONCLUSIONS

Kalman filters show a great deal of promise when tracking expected blog content. Given the prior work in the domain of topic tracking, the application of Kalman Filters to change tracking of blog data is a natural extension. We found anecdotal

evidence that Kalman filters can be used to accurately flag the unexpected change in a blog's content using a measure of deviation in error. While the prediction of a future version's composition is very error-prone, significantly large deviations from the predictions were empirically determined to reflect changes in page behavior and content we felt were unexpected.

8. REFERENCES

[1] H. Ashman. *Electronic document addressing: dealing with change*. ACM Comput. Surv., 32(3):201–212, 2000.

[2] I. Askehave and A. E. Nielsen. *What are the characteristics of digital genres? - genre theory from a multi-modal perspective.* In Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05), volume 04, page 98a, Los Alamitos, CA, USA, 2005. IEEE Computer Society.

[3] D. M. Blei, and J. D. Lafferty. *Dynamic topic models*. In Proceedings of the 23rd international Conference on Machine Learning pages 113-120, New York, NY, ACM Press.

[4] E. S. Boese and A. E. Howe. *Effects of web document evolution on genre classification.* In CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management, pages 632–639, New York, NY, USA, 2005. ACM Press.

[5] P. L. Bogen, L. Francisco-Revilla, R. Furuta, T. Hubbard, U. P. Karadkar, and F. Shipman. *Longitudinal study of changes in blogs.* In JCDL '07: Proceedings of the 2007 conference on Digital libraries, pages 135–136, New York, NY, USA, 2007. ACM Press.

[6] A. Chin and M. Chignell. *A social hypertext model for finding community in blogs*. In HYPERTEXT '06: Proceedings of the seventeenth conference on Hypertext and hypermedia, pages 11–22, New York, NY, USA, 2006. ACM Press.

[7] G. Cselle, K. Albrecht, K., and R. Wattenhofer. *BuzzTrack: topic detection and tracking in email.* In Proceedings of the 12th international Conference on intelligent User interfaces, Honolulu, Hawaii, 2007. ACM Press.

[8] A. Dalli. *System for spatio-temporal analysis of online news and blogs.* In WWW '06: Proceedings of the 15th international conference on World Wide Web, pages 929–930, New York, NY, USA, 2006. ACM Press.

[9] M. Y. Ivory, and R. Megraw. *Evolution of web site design patterns*. ACM Trans. Inf. Syst. 23, 4 (2005), 463–497.

[10] A. Krause, J. Leskovec, and C. Guestrin. *Data association for topic intensity tracking*. In Proceedings of the 23rd international Conference on Machine Learning, Pittsburgh, Pennsylvania, 2006. ACM Press.

[11] W. Koehler. *Web page change and persistence—a four-year longitudinal study*. J. Am. Soc. Inf. Sci. Technol. 53, 2 (2002), 162–171.

[12] P. Maybeck. *Stochastic models, estimation, and control.* Vol. 141. Series: Mathematics in Science and Engineering, 1979.

[13] A. K. McCallum. Mallet: *A machine learning for language toolkit.* http://mallet.cs.umass.edu, 2002.

[14] M. Porter. *The english (porter2) stemming algorithm.* Available at: http://snowball.tartarus.org/algorithms/english/stemmer.html, September 2002.

[15] B. Starr, M. S. Ackerman, and M. Pazzani. *Do-i-care: a collaborative web agent.* In CHI '96: Conference companion on Human factors in computing systems, pages 273–274, New York, NY, USA, 1996. ACM Press

[16] G. Welch and G. Bishop. *An introduction to the Kalman filter.* Available at: http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf, 2006.

[17] B. Van Durme, Y. Huang, A. Kupść, and E. Nyberg. *Towards light semantic processing for Question Answering.* In Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning - Volume 9, pages 54-61, Morristown, NJ, 2003, Human Language Technology Conference. Association for Computational Linguistics.

[18] X. Wang and A. McCallum. *Topics over time: a non-Markov continuous-time model of topical trends.* In Proceedings of the 12th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining, pages 424-433, Philadelphia, PA, 2006. ACM Press.