Text Classification Kernels for Quality Prediction over the C3 Data Set*

Bálint Daróczy Dávid Siklósi Róbert Pálovics András A. Benczúr
Institute for Computer Science and Control, Hungarian Academy of Sciences (MTA SZTAKI)
{daroczyb, sdavid, rpalovics, benczur}@ilab.sztaki.hu

ABSTRACT

We compare machine learning methods to predict quality aspects of the C3 dataset collected as a part of the Reconcile project. We give methods for automatically assessing the credibility, presentation, knowledge, intention and completeness by extending the attributes in the C3 dataset by the page textual content. We use Gradient Boosted Trees and recommender methods over the evaluator, site, evaluation triplets and their metadata and combine with text classifiers. In our experiments best results can be reached by the theoretically justified normalized SVM kernel. The normalization can be derived by using the Fisher information matrix of the text content. As the main contribution, we describe the theory of the Fisher matrix and show that SVM may be particularly suitable for difficult text classification tasks.

Categories and Subject Descriptors

H.3 [**Information Systems**]: Information Storage and Retrieval; I.2 [**Computing Methodologies**]: Artificial Intelligence; I.7.5 [**Computing Methodologies**]: Document Capture—*Document analysis*

General Terms

Kernel Methods, Document Classification, Information Retrieval

Keywords

Web Quality, Credibility, Machine Learning, Fisher Information Matrix

*The publication was supported in part by the EC FET Open project "New tools and algorithms for directed network analysis" (NADINE No 288956), by the Momentum Grant of the Hungarian Academy of Sciences, by OTKA NK 105645, the KTIA_AIK_12-1-2013-0037 and the PIAC_13-1-2013-0197 projects. The projects are supported by Hungarian Government, managed by the National Development Agency, and financed by the Research and Technology Innovation Fund. Work conducted at the Technical University Budapest has been developed in the framework of the project "Talent care and cultivation in the scientific workshops of BME" project. This project is supported by the grant TÁMOP - 4.2.2.B-10/1–2010-0009.

WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2742126.

1. INTRODUCTION

Mining opinion from the Web and assessing its quality and credibility became a well-studied area [9]. Known results typically mine Web data on the micro level, analyzing individual comments and reviews. Recently, several attempts were made to manually label and automatically assess the credibility of Web content [19, 21]; among others, Microsoft created a reference data set [27]. Classifying various aspects of quality on the Web host level were, to our best knowledge, first introduced as part of the ECML/PKDD Discovery Challenge 2010 tasks [28].

Classification for quality aspects of Web pages or hosts turned out to be very hard. For example, the ECML/PKDD Discovery Challenge 2010 participants stayed with AUC values near 0.5 for classifying trust, bias and neutrality. Later we were able to slightly improve their results and our best performance has only slightly extended the AUC of 0.6 [28]. Since these attributes constitute key aspects of Web quality, our goal is to improve the classification techniques for these tasks.

In this paper we address the WebQuality 2015 Data Challenge by comparing prediction methods for the C3 data set. The data set was created in the Reconcile[1] project and contains 22325 evaluations (five dimensions, among them credibility) of 5704 pages given by 2499 people. The mTurk platform were used for collecting evaluations.

In our earlier findings on different Web spam and quality corpora [12], the bag-of-words classifiers based on the top few 10,000 terms performed best. We were able to significantly improve the traditional Web spam features [5] similar to the C3 attributes. In this paper our main goal is to evaluate known methods and combine them with new means of text classification particularly suited to the quality related tasks in question.

While we are aware of no other results over the C3 data set, we collect reference methods from Web credibility research results. Existing results fall in four categories: Bag of Words; language statistical, syntactic, semantic features; numeric indicators of quality such as social media activity; and assessor-page based collaborative filtering.

User and page-based collaborative filtering is suggested in [21] in combination with search engine rankings. We reuse our RecSys Challenge 2014 second place winner solution [20] to build a strong baseline method over the evaluator, site, evaluation triplets including the evaluator and site side information.

[1] `http://reconcile.pjwstk.edu.pl/`

Social media and network based features appear already for Web spam [5, 15]. In a collection designed similar to C3 [19], social and general popularity and linkage were introduced and used for credibility assessment. Some of these features, in particular social media popularity, are used by the RecSys Challenge 2014 [20] as well and hence we deploy the methods we used there.

Content statistics as a concise summary that may replace the actual terms in the document were introduced first in the Web spam research [5]. The C3 data set includes content quality and appearance features described among others in [19].

In order to perform text classification, we crawled the pages listed in the C3 data set. By using the bag of words representation of the Web page content, our goal is to combine all above methods with known and new kernel based text classifiers. Our classifier ensemble consists of the following components:

- Gradient Boosted Trees and recommender methods that reached us second place at the RecSys Challenge 2014 [20].
- Standard text classifiers, including our biclustering based method that performed best over the DC2010 data set [28].
- A new similarity kernel based SVM on the Fisher Information Matrix that may work over arbitrarily defined similarity measures over pairs of pages, using not only the text but also the C3 attributes.

Our best results reach the AUC of 0.74 for credibility, 0.81 for Presentation, 0.70 for Knowledge, 0.71 for Intentions and 0.70 for Completeness. We may hence say that all results reach the level of practical usability. Text classification is the main component: alone it reaches 0.73, 0.77, 0.69, 0.71 and 0.70, respectively, for the five quality dimensions.

The rest of this paper is organized as follows. First we begin with an extended motivation of our new text classification technique. After listing related results, in Section 2 we describe the data set used in this paper. In Section 3 we describe our classification framework. The results of the classification experiments over the C3 data set can be found in Section 4.

1.1 Motivation

In our new similarity kernel method, our goal is to move from terms as features to content similarity as features. On one hand, content similarity is more general and it can be defined by using the attributes other than term frequencies as well. Similarity based description is also scalable since we may select the number of reference documents as large as it remains computationally feasible.

In the paper our main goal is to define a theoretically justified kernel function over Web page similarities defined in a general way. Similarity may be based on the distribution of terms, the distance in the numeric C3 data attributes, or distances from clusters as we defined in [28].

By considering general notions of similarity as object descriptors for classification, we may combine different modalities in a theoretically justified way too. For example, kernel selection methods [23] performed well for image classification tasks [8] but kernel fusion methods from [23] have a very large number of parameters that are difficult to learn.

In our new method, we consider the similarity of a Web page in question to a set of selected reference pages as a generative model. By assuming independence of the reference pages, the generative model can be computed as we will describe in Section 3.2. Hence we may obtain theoretically justified coefficients to weight the importance of the different similarity functions and reference Web pages.

1.2 Related Results

Web users usually lack evidence about author expertise, trustworthiness and credibility [5]. The first results on automatic Web quality classification focus on Web spam. In the area of the so-called Adversarial Information Retrieval workshop series ran for five years [13] and evaluation campaigns, the Web Spam Challenges [4] were organized. The ECML/PKDD Discovery Challenge 2010 extended the scope by introducing labels for genre and in particular for three quality aspects [28].

Our baseline classification procedures are collected by analyzing the results of the Web Spam Challenges and the ECML/PKDD Discovery Challenge 2010. In our previous work [11, 28], we improved over the best results of the participants by using new text classification methods.

Recent results on Web credibility assessment [19] use content quality and appearance features combined with social and general popularity and linkage. After feature selection, they use 10 features of content and 12 of popularity by standard machine learning methods of the scikit-learn toolkit.

If sufficiently many evaluators assess the same Web page, one may consider evaluator and page-based collaborative filtering [21] for credibility assessment. In this setting, we face a dyadic prediction task where rich metadata is associated with both the evaluator and especially with the page. The Netflix Prize competition [3] put recommender algorithms through a systematic evaluation on standard data [2]. The final best results blended a very large number of methods whose reproduction is out of the scope of this experiment. Among the basic recommender methods, we use matrix factorization [17, 29]. In our experiments we use the factorization machine [24] as a very general toolkit for expressing relations within side information. Recently, the RecSys Challenge 2014 run a similar dyadic prediction task where Gradient Boosted Trees [30] performed very well [20].

2. THE DATA SET

The C3 data set consists of 22325 Web page evaluations in five dimensions (credibility, presentation, knowledge, intentions, completeness) of 5704 pages given by 2499 people. Ratings are similar to the dataset built by Microsoft for assessing Web credibility [27], on a scale of four values 0-4, with 5 indicating no rating. The distribution of the scores for the five evaluation dimensions can be seen in Fig. 1. Since multiple values may be assigned to the same aspect of a page, we simply average the human evaluations per page. We may also consider binary classification problems by assigning 1 for above 2.5 and 0 for below 2.5.

Since earlier results [21] suggest the use of collaborative filtering along the page and evaluator dimensions, we measure the distribution of the number of evaluations given by the same evaluator and for the same site in Fig. 2.

Distribution of the variance of the ratings is shown by heatmap of all pairs of ratings given for the same page and same dimension by pairs of different evaluators in Fig. 3.

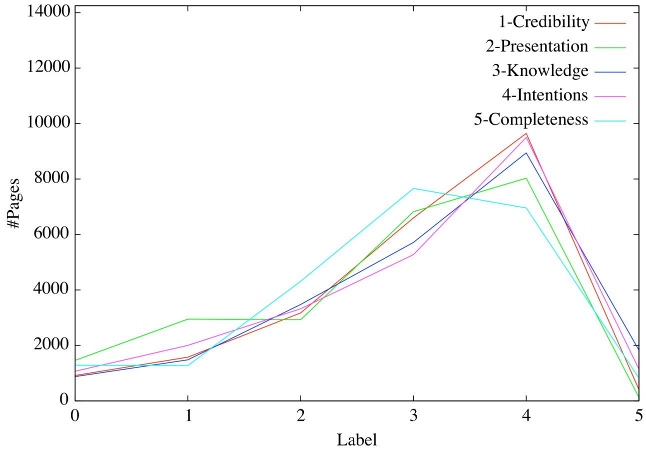

Figure 1: The distribution of the scores for the five evaluation dimensions.

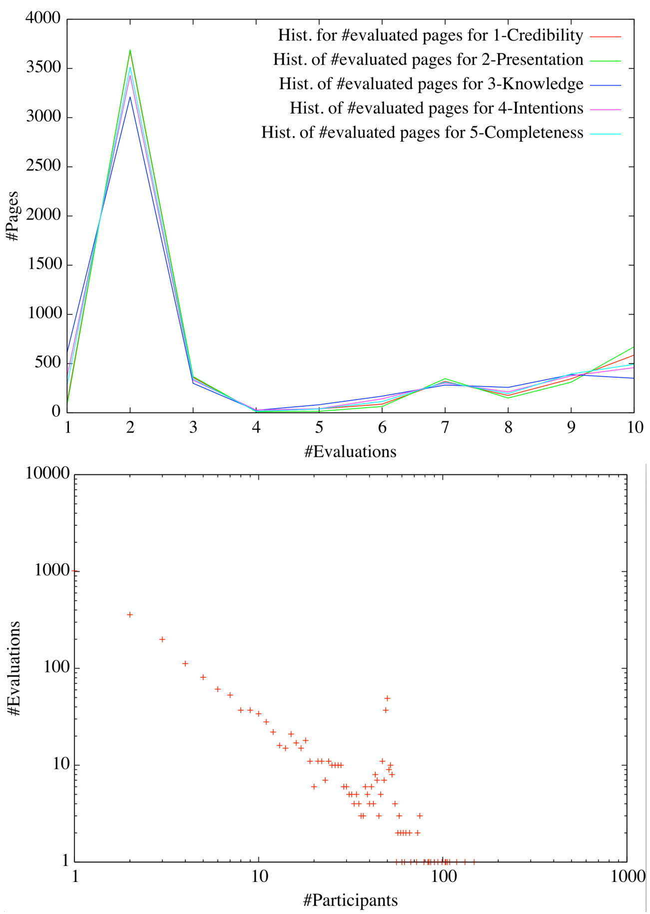

Figure 2: The distribution of the number of evaluations given by the same evaluator (top) and for the same site (bottom).

Note that 65% of the C3 URLs returned OK HTTP status but 7% of them could no longer be crawled. Redirects reached over 20% that we followed and substituted for the original page.

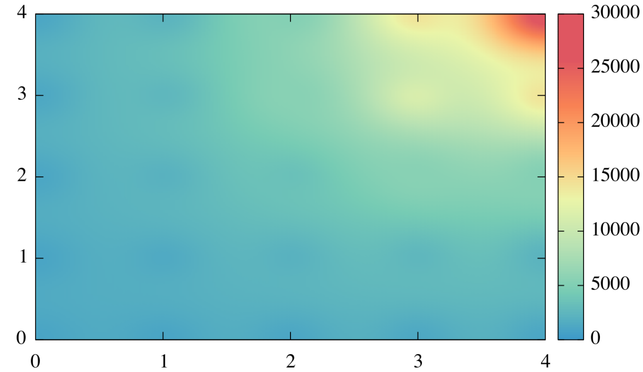

Figure 3: The number of pairs of ratings given by different assessors for the same aspect of the same page.

3. CLASSIFICATION FRAMEWORK

In this section we enumerate the methods we combine for assessing the five quality aspects. The C3 data set contains numeric attributes for the evaluator, the page, and the evaluation itself, which can be considered as triplets in a recommender system. The majority of the evaluators however rated only one Web page and hence we expect low performance of the recommender methods over this data set. Most important elements of our classifier ensemble will hence use the bag of words representation of the page content.

3.1 SVM over bag of words

The classification power of Support Vector Machine [7] over bag of words representations has been shown in [1, 5]. The models rely on term and inverse document frequency values (TF and IDF): aggregated as TF.IDF and BM25. The BM25 scheme turned out to perform best in our earlier results [11, 28], where we applied SVM with various linear and polynomial kernel functions and their combinations.

3.2 New method: Fisher Kernel over similarities

A natural idea to handle distances of pairs of observation is to use kernel methods. A kernel acts as an inner product between two observations in certain large dimensional space where Support Vector Machine, a form of a high dimensional linear classifier, can be used to separate the data points [26]. Under certain mathematical conditions, we have a freedom to define the kernel function by giving the formula for each pair of observations.

In order to combine the textual and C3 data attributes for kernel based classification and regression, we use a linear kernel support vector machine over distances from a selected set of reference pages as described in [8].

Given a sample R of the Web pages, we define a generative model where testing pages are characterized based on their similarity to samples in R. By Jaakkola and Haussler [16], generative models have a natural kernel function based on the Fisher information matrix F:

$$K_{Fisher}(X, Y) = G_X^T F^{-1} G_Y, \qquad (1)$$

where G_X and G_Y are the gradient vectors (Fisher score) derived from the underlying generative model. The Fisher kernel can be translated into a linear kernel function using

Cholesky decomposition of the Fisher information matrix. We will refer the normalized Fisher score as Fisher vector: $F_X = G_x F^{-\frac{1}{2}}$. In our experiments we approximate the Fisher information matrix with the diagonal as suggested in [16].

Next we sketch the steps of deriving that the Fisher matrix based distance is simply the Euclidean distance over the $K \cdot |R|$ dimensional vector of the similarity to pages in R with K representations.

In the generative model of pages based on the similarity to pages in the sample R, our factor graph is a star that consists of the pairs of x connected to the elements $r \in R$. We think of our graph as a Markov Random Field over the samples. By the Hammersley–Clifford theorem [25] our joint distribution has a form of

$$p(x \mid \Omega) = \frac{\exp(-U(x \mid \Omega))}{Z}, \tag{2}$$

where Z is a normalizing constant and Ω is the set of parameters of our joint distribution. We define our energy function as

$$U(x \mid \Omega = \{\alpha\}) = \sum_{r \in R} \sum_{k=1}^{K} \alpha_{rk} \mathrm{dist}_k(x, x_r), \tag{3}$$

where K is the number of different distance functions and $\Omega = \{\alpha_{rk}\}$ is the set of the parameters.

It can be shown that the Fisher information matrix is simply the normalized variance matrix of the joint distribution $\mathrm{dist}_k(x, x_r)$ for $r \in R$, i.e. the Fisher kernel is the linear kernel over the normalized distances. In the Fisher kernel $\alpha_r k$ cancel out in the derivatives. The mean and the variance of $\mathrm{dist}_k(x, x_r)$ can be approximated by the training data.

The dimensionality of the Fisher vector (the normalized Fisher score) equals with the size of the parameter set of our joint distribution, in our case it depends only on the size of the reference set and the number of representations, $K \cdot |R|$.

Since kernel methods are feasible for regression [22, 26], we also use the methods of this subsection for predicting the numeric evaluation scores.

3.3 Biclustering

We overview the method that performed best for assessing the quality aspects of the DC2010 data [28]. We use Dhillon's information theoretic co-clustering algorithm [10] to cluster pages and terms simultaneously. Important to note that unlike in the original method [10] that uses Kullback-Leibler divergence, we use Jensen-Shannon, the symmetric version in the biclustering algorithm that makes very large difference in classification quality.

In [28] we describe pages by distances from page clusters. To exploit the Fisher kernel we can think of this page clusters as additional samples with a specific distance function. This results sparsity in our previously defined energy function

$$U(x \mid \Omega = \{\alpha, \beta\}) = U(x \mid \Omega = \{\alpha\}) + \sum_{C_i \in C} \beta_i \mathrm{dist}(x, C_i), \tag{4}$$

where C_i corresponds to the ith cluster, therefore the clusters behave as a secondary sample set to R on a cost of expanded dimension.

3.4 Gradient Boosted Trees and Matrix factorization

We apply Gradient Boosting Trees [30] and matrix factorization on the user and C3 data features. We used two different matrix factorization techniques. The first one is a traditional matrix factorization method [17], while the second one is a simplified version of Steffen Rendle's LibFM algorithm [24]. Both techniques use stochastic gradient descent to optimize for mean-square error on the training set. LibFM is particularly designed to use the side information of the evaluators and the pages.

3.5 Evaluation metrics

First, we consider binary classification problems by simply averaging the human evaluations per page and assign them 1 for above 2.5 and 0 for below 2.5. The standard evaluation metrics since the Web Spam Challenges [4] is the area under the ROC curve (AUC) [14]. The use of Precision, Recall and F are discouraged by experiences of the Web spam challenges.

Unlike spam classification, the translation of quality assessments into binary values is not so obvious. We also test regression methods evaluated by Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).

4. RESULTS

In this section we measure the accuracy of various methods and their combinations. The detailed results are in Table 1, in four groups. The first group gives the baseline methods. Below, we apply the similarity kernel separate for the corresponding attributes. In the third group we combine multiple similarity functions by the similarity kernel. Finally, in the last group, we average after standardizing the predictions. In Table 2 part of the methods are tested for regression.

4.1 C3 data attributes

For user and item features we experiment with GraphLab Create[2] [18] implementation of Gradient Boosted Tree and matrix factorization techniques. In case of the gradient boosted tree algorithm (GBT) we set the maximum depth of the trees 4, and enabled maximum 18 iterations. To determine the advantage of additional side information over the original matrix factorization technique (MF) we use factorization machine (LibFM) for user and item feature included collaborative filtering prediction. As seen from the tables, matrix factorization (MF) fails due to the too low number of ratings by user and by document but LibFM can already take advantage of the website metadata with performance similar to GBT.

4.2 Linear kernel SVM

Our Bag of words models use the top $30k$ stemmed terms. For TF, TF.IDF and BM25, we show results for linear kernel SVM as it outperforms the RBF and polynomial kernels. We use LibSVM [6] for classification the Weka implementation of SMOReg [22] for regression.

4.3 Fisher kernel methods

The similarity kernel described in Section 3.2 gives the best results both for classification and for regression. For

[2] `http://graphlab.com/products/create/`

Method	Credi-bility	Presen-tation	Know-ledge	Inten-tions	Complete-ness	Avg
Gradient Boosted Tree (**GBT**)	0.6492	0.6558	0.6179	0.6368	0.7845	0.6688
Factorization Machine (**LibFM**)	0.6563	0.6744	0.6452	0.6481	0.7234	0.6695
Marix Factorization (**MF**)	0.5687	0.5613	0.5966	0.5700	0.5854	0.5764
TF linear kernel	0.6484	0.6962	0.6239	0.6767	0.6205	0.6531
TF.IDF linear kernel	0.6571	0.7020	0.5935	0.6824	0.6128	0.6496
BM25 linear kernel (**Lin**)	0.7236	0.7480	0.6278	0.6987	0.6633	0.6923
Bicluster linear kernel	0.6402	0.7467	0.5796	0.6482	0.6382	0.6506
Bicluster Sim kernel	0.6744	0.7718	0.6379	0.6830	0.6560	0.6846
C3 attributes Sim kernel	0.6267	0.7706	0.6327	0.6408	0.6149	0.6571
TF J–S Sim kernel	0.6902	0.7404	0.6758	0.7047	0.6778	0.6978
TF L_2 Sim kernel	0.6335	0.6882	0.6200	0.6585	0.6300	0.6460
TF.IDF J–S Sim kernel	0.7006	0.7546	0.6552	0.7073	0.6791	0.6994
TF.IDF L_2 Sim kernel	0.6461	0.7152	0.6013	0.6902	0.6353	0.6576
BM25 J–S Sim kernel	0.6956	0.7473	0.6351	0.6529	0.6222	0.6706
BM25 L_2 Sim kernel	0.7268	0.7715	0.6741	0.7081	0.6898	0.7141
BM25 L_2 & J–S Sim kernel (**BM25**)	0.7313	0.7761	0.6926	0.7141	0.7003	0.7229
BM25 & C3 Sim kernel	0.7449	0.8029	0.7009	0.7148	0.6993	0.7326
BM25 & Bicluster & C3 (**All**) Sim kernel	0.7457	0.8086	0.7063	0.7158	0.7052	0.7363
Lin + GBT	0.7296	0.8056	0.6589	0.6783	0.6939	0.7133
Lin + LibFM	0.7400	0.7769	0.6622	0.6733	0.6975	0.7100
All Sim kernel + Lin + GBT	0.7549	0.8179	0.6916	0.7098	0.7123	0.7373

Table 1: Detailed performance over the C3 labels in terms of AUC

Method		Credi-bility	Presen-tation	Know-ledge	Inten-tions	Complete-ness	Avg
Gradient Boosted Tree (GBT)	MAE	1.5146	1.3067	1.2250	1.2737	1.4438	1.3528
	RMSE	1.6483	1.4510	1.3658	1.4132	1.6021	1.4961
Factorization Machine (LibFM)	MAE	1.5313	1.3213	1.2303	1.2632	1.4984	1.3689
	RMSE	1.6725	1.4745	1.3744	1.4073	1.6759	1.5209
Matrix Factorization (MF)	MAE	1.7450	1.4093	1.3676	1.2905	1.5794	1.4784
	RMSE	1.9174	1.5912	1.5540	1.4636	1.7583	1.6569
BM25 linear kernel (Lin)	MAE	0.5562	0.7230	0.6052	0.5979	0.5896	0.6144
	RMSE	0.7085	0.9072	0.7784	0.7910	0.7724	0.7915
BM25 L_2 Sim kernel	MAE	0.5678	0.7083	0.6228	0.5946	0.6045	0.6196
	RMSE	0.7321	0.9307	0.8038	0.7878	0.7930	0.8095
Bicluster Sim kernel	MAE	0.5340	0.6868	0.6039	0.5883	0.5813	0.5989
	RMSE	0.6958	0.8906	0.7861	0.7778	0.7624	0.7825
BM25 & Bicluster & C3 All Sim kernel	MAE	0.5403	0.6324	0.5946	0.5952	0.5829	0.5891
	RMSE	0.7106	0.8357	0.7763	0.7879	0.7661	0.7753

Table 2: Detailed performance over the C3 labels in terms of RMSE and MAE

distance, we use L_2 for the C3 attributes as well as TF, TF.IDF and BM25. For the last three, we also use the Jensen–Shannon divergence (J–S) as we suggested in [28]. While the similarity kernel over the bicluster performs weak for classification, it is the most accurate single method for regression.

In the similarity kernel, we may combine multiple distance measures by Equation (3). The All Sim method fuses four representations: J–S and L_2 over BM25 and L_2 for C3 and the bicluster representation. By the linearity of the Fisher kernel, we may use LibSVM [6] for classification and SMOReg [22] for regression.

4.4 Classifier ensembles

Without using the similarity kernel, the best method is the average of the linear kernel over BM25 (Lin) and GBT. The performance is similar to the BM25 L_2 similarity kernel. As a remarkable feature of the similarity kernel, we may combine multiple distance functions in a single kernel. The best method (All Sim) outperforms the best combination not using the similarity kernel (Lin + GBT) by 3.2%. The difference is 7.2% for classifying "knowledge". The same method performs bests for regression too.

The similarity kernel method can also resist noise and learn from small training sets. If we add 10% noise in the training set, the combination of all similarity kernels deteriorates only to an average AUC of 0.7241 from 0.7363 (1.7%). In contrast, the best BM25 SVM result 0.6923 degrades to

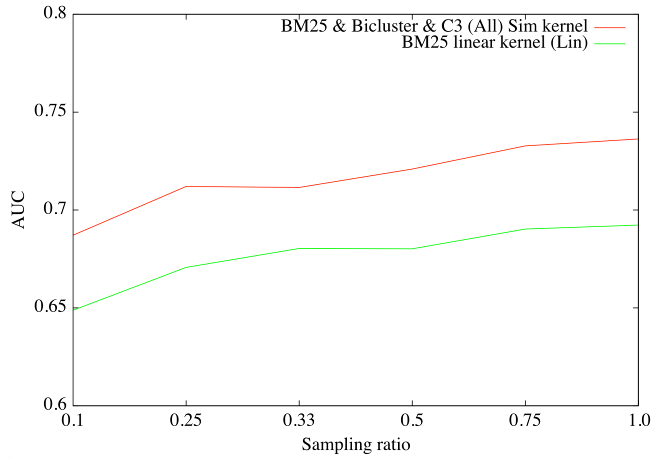

Figure 4: AUC as the function of the size of the training set, given as percent of the full3040, for the baseline BM25 with linear kernel and All with similarity kernel.

0.6657 (3.85%), both with variance 0.004 for ten independent samples. The robustness of the similarity kernel for small training sets is similar to BM25 with linear kernel, as seen in Fig. 4.

5. CONCLUSIONS

Over the C3 data sets, we gave a large variety of methods to predict quality aspects of Web pages, including collaborative filtering and methods that use evaluator and page metadata as well as the content of the page. We achieved best performance by our theoretically justified kernel method over the content of the page and C3 attributes. Our results are promising in that our AUC is stable over 0.7 for all aspects with "presentation" surpassing 0.8. The support vector regression methods also perform with error less than one on the range of 0–4.

6. REFERENCES

[1] J. Abernethy, O. Chapelle, and C. Castillo. WITCH: A New Approach to Web Spam Detection. In *Proc. 4th AIRWeb*, 2008.

[2] R. M. Bell and Y. Koren. Lessons from the netflix prize challenge. *ACM SIGKDD Explorations Newsletter*, 9(2):75–79, 2007.

[3] J. Bennett and S. Lanning. The netflix prize. In *KDD Cup and Workshop in conj. KDD 2007*, 2007.

[4] C. Castillo, K. Chellapilla, and L. Denoyer. Web spam challenge 2008. In *Proc. 4th AIRWeb*, 2008.

[5] C. Castillo, D. Donato, L. Becchetti, P. Boldi, S. Leonardi, M. Santini, and S. Vigna. A reference collection for web spam. *SIGIR Forum*, 40(2):11–24, December 2006.

[6] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[7] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20, 1995.

[8] B. Z. Daróczy, D. Siklósi, and A. Benczúr. SZTAKI @ ImageCLEF 2012 Photo Annotation. In *Working Notes of the ImageCLEF 2012 Workshop*, 2012.

[9] K. Dave, S. Lawrence, and D. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proc. 12th WWW*, pages 519–528. ACM, 2003.

[10] I. Dhillon, S. Mallela, and D. Modha. Information-theoretic co-clustering. *Proc. 9th SIGKDD*, pages 89–98, 2003.

[11] M. Erdélyi, A. A. Benczúr, B. Daróczy, A. Garzó, T. Kiss, and D. Siklósi. The classification power of web features. *Internet Mathematics*, 10(3-4):421–457, 2014.

[12] M. Erdélyi, A. Garzó, and A. A. Benczúr. Web spam classification: a few features worth more. In *WebQuality 2011*. ACM Press, 2011.

[13] D. Fetterly and Z. Gyöngyi. Proc. 5th AIRWeb. 2009.

[14] J. Fogarty, R. S. Baker, and S. E. Hudson. Case studies in the use of roc curve analysis for sensor-based estimates in human computer interaction. In *Proc. Graphics Interface*, pp. 129–136, 2005.

[15] Z. Gyöngyi and H. Garcia-Molina. Spam: It's not just for inboxes anymore. *IEEE Computer Magazine*, 38(10):28–34, October 2005.

[16] T. S. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. *Advances in NIPS*, pp. 487–493, 1999.

[17] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[18] Y. Low, D. Bickson, J. Gonzalez, C. Guestrin, A. Kyrola, and J. M. Hellerstein. Distributed graphlab: a framework for machine learning and data mining in the cloud. *Proc VLDB*, 5(8):716–727, 2012.

[19] A. Olteanu, S. Peshterliev, X. Liu, and K. Aberer. Web credibility: Features exploration and credibility prediction. In *Advances in Information Retrieval*, pages 557–568. Springer, 2013.

[20] R. Pálovics, F. Ayala-Gómez, B. Csikota, B. Daróczy, L. Kocsis, D. Spadacene, and A. A. Benczúr. Recsys challenge 2014: an ensemble of binary classifiers and matrix factorization. In *Proc. Recommender Systems Challenge*, page 13. ACM, 2014.

[21] T. G. Papaioannou, J.-E. Ranvier, A. Olteanu, and K. Aberer. A decentralized recommender system for effective web credibility assessment. In *Proc. 21st CIKM*, pp. 704–713. ACM, 2012.

[22] J. C. Platt. Sequential minimal optimization: A fast algorithm for training support vector machines. Technical report, ADVANCES IN KERNEL METHODS - SUPPORT VECTOR LEARNING, 1998.

[23] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet. simplemkl. *JMLR*, 9:2491–2521, 2008.

[24] S. Rendle, Z. Gantner, C. Freudenthaler, and L. Schmidt-Thieme. Fast context-aware recommendations with factorization machines. In *Proc. 34th SIGIR*, pp. 635–644. ACM, 2011.

[25] B. D. Ripley and F. P. Kelly. Markov point processes. *Journal of the London Mathematical Society*, 2(1):188–192, 1977.

[26] B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors. *Advances in kernel methods: support vector learning*. MIT Press, Cambridge, MA, USA, 1999.

[27] J. Schwarz and M. Morris. Augmenting web pages and search results to support credibility assessment. In *Proc. SIGCHI*, pp. 1245–1254. ACM, 2011.

[28] D. Siklósi, B. Daróczy, and A. Benczúr. Content-based trust and bias classification via biclustering. In *Proc. Webquality*, pp. 41–47. ACM, 2012.

[29] G. Takács, I. Pilászy, B. Németh, and D. Tikk. Investigation of various matrix factorization methods for large recommender systems. In *Proc. 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition*, pages 1–8. ACM, 2008.

[30] Z. Zheng, H. Zha, T. Zhang, O. Chapelle, K. Chen, and G. Sun. A general boosting method and its application to learning ranking functions for web search. In *Advances in NIPS*, pp. 1697–1704, 2008.