Empirical Bayes Screening for Link Analysis *

Anna Goldenberg, Andrew Moore
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA
{anya,awm}@cs.cmu.edu

Abstract

The domain of link analysis has recently re-ignited interest among researchers due to its applicability to new areas such as intelligence analysis (for example, identifying cliques of suspicious people), large scale social network analysis and genomics. The area of link analysis is not new and comprise a number of techniques developed by different communities. In this paper we propose a statistical approach to answering questions such as: what would be the "interesting" k-tuples of entities (that can be people, ingredients in a recipe, etc - depending on the application), given a dataset of observed n-tuples of entities. A typical example of an n-tuple might be a set of people observed to be having a meeting, or observed traveling to the same destination. Currently, it is common to work with pairwise count matrices. Empirical Bayes Screening (EBS) has several advantages over existing methods, one of them being the ability to take advantage of the interactions of higher order (for example, a group of three people significantly working together even though no two of them have significantly atypical pairwise interaction). EBS has the additional advantage of being insensitive to the small sample size of co-occurrences. We discuss advantages and disadvantages of the algorithm and provide performance analysis based on several datasets.

1 Introduction

There are a variety of algorithms in the domain of social network analysis stemming from graph theory and other fields that search for linkages among entities. Most traditional approaches create "links" based on the pairwise linkage information. In this paper, we search for triples, quadruples, up to k-tuples, that could not be explained by the relationships of the lower degree.

For example, suppose person A is a known criminal and we want to find who are the collaborators of A. Imagine that B, C and D are the true collaborators of A. In our dataset we have information about thousands of meetings of different people. A, B, C and D have participated in quite a few meetings separately but have occurred together relatively few times. Just by chance there could be other people who have independently co-occurred with A more times than B, C or D, so when looking at counts of pairwise occurrences, pairs AB, AC and AD would not be ranked very high. In fact, there could be pairs of people that have met with A more often then BC, CD or BD, however when looking at the quadruples, we could notice that A,B,C and D have occurred consistently together, whereas for other random triplets that would not be the case. This is a situation where the pairwise independence assumption would fail. The described scenario is quite likely in the real datasets. The importance of detecting the true collaborators by fitting appropriate models is evident from this toy example.

The statistical algorithm we are proposing to use here has been developed at AT&T by Bill DuMouchel [DuMouchel, 1999] and applied to the market-basket analysis by DuMouchel and Pregibon [DuMouchel and Pregibon, 2001]. In this paper we show that it is well suited for searches of significant k-tuples in the general social network type setting. We discuss advantages and disadvantages of using this method on real data and show the performance of the algorithm in applications to simulated as well as real world datasets.

2 Approach

There are several possible approaches to the search of significant tuples of high degree of interaction. The methodology we propose involves loglinear models and Empirical Bayes shrinkage. It might seem unnecessarily complicated, so in the following sections we explain the pitfalls of using simpler approaches.

2.1 Frequency counting

The most naive approach would be to rank the k-tuples based on their frequency. In other words, we can count the number of occurrences for each tuple of size k and return the top most-frequent tuples. There are several problems with this approach. This method will be adequate when the entities in the most frequent tuple only occur together and never separately. In this case, the k-tuple might correctly look suspicious. However, it is also possible that the subsets of the tuple happen to be promiscuous and occur in many links. Clearly,

*This research is supported by DARPA under award number F30602-01-2-0569.

those two cases should be treated very differently, but the *frequency counting* approach would not be able to differentiate between the two. There are other drawbacks to this approach, but the lack of robustness described above is a major fault rendering this approach unusable in the situations where the accuracy is an issue.

2.2 Estimates based on complete independence, rank based on ratio of true count to its estimate

To combat the problem of differentiating between the tuples in the above case, we will first compute estimates and then look at the ratios $r_{ij..k} = n_{ij..k}/e_{ij..k}$, where $n_{ij..k}$ is the true number of occurrences of the tuple $\{ij..k\}$ and $e_{ij..k}$ is the corresponding estimate. Lets suppose that entities $i, j..k$ have occurred in links independently from each other, i.e. assume complete independence. Clearly, the problem of promiscuity in this case is solved: the entities that have a high probability of occurring in the dataset, have a higher probability of randomly forming k-tuples and those k-tuples will not be ranked highly, since their estimates e will also be high, thus making the ratios r smaller. Suppose however, we have three k-tuples that have exactly the same ratios (n:e): $300 : 30$, $30 : 3$ and $3 : 0.3$. The ranking based on ratio r will not distinguish between those 3 tuples, however they clearly cannot be treated in the same way. To alleviate this problem we introduce the next step.

2.3 Empirical Bayes

In the majority of the real world situations, the most interesting tuple in the example above is not the tuple that has the largest frequency (noted in the first example). It is also not the one with the lowest frequently, simply because the sample size is too tiny in an otherwise large dataset to make any valid statistical inference. In fact, the small sample size invalidates another algorithm discussed in Related Work (Section 7). The tuple of interest will be a compromise. Given a choice between $300 : 30$, $30 : 3$ and $3 : 0.3$, the first ($300 : 30$) may turn out to be not interesting due to the promiscuity of some of its members and the last ($3 : 0.3$) may not be statistically significant. What scoring measure can be used to then prefer the middle ($30 : 3$)? The answer is well understood in the statistics literature: smoothing. In this paper, we use smoothing by Empirical Bayes analysis.

Empirical Bayes methodology allows for a significant improvement over naive scoring functions (such as ratio $r = n/e$), due to it's "shrinking" property. If there is a small support for the k-tuple (i.e. not too many co-occurrences in the data), then even if the independence estimate is very small, statistically we cannot say much about the data, hence measure r, will be "shrunk" to the prior distribution, making this methodology more robust when the frequencies of k-tuples have high variance. However, there is still a problem with this approach. The complete independence model might not be a good fit to the data and even though our new method of ranking is more robust, the parameters of the model that we calculate using Empirical Bayes will be invalid, since the idea of Empirical Bayes is to estimate the parameters from data.

2.4 Fitting loglinear models to calculate estimates

It is obvious that the complete independence assumption is rather naive. In the real world, people don't randomly go to meetings. Usually the majority of the people who are present at the meeting collaborate on a particular project, are friends or have common interests. It is possible that some people appear at meetings randomly, but the underlying structure is usually somewhat more complicated than naive independence. Thus, we would like to use models that are capable of describing more complex interactions between entities in a k-tuple. A general apparatus for describing any type of interaction between items in a set is a loglinear model. For example, consider a triple ABC. We can describe the mutual independence of all pairs that are subsets of the given triple by including terms $u_{AB} + u_{BC} + u_{AC}$ in the given model, but not including the saturated term u_{ABC}, which describes the three way dependence between entities. Terms such as u_{AB} describe the interaction between items A and B. This is precisely what we need in order to describe more complicated relations between the entities that constitute the tuple. In fact, if a simpler model could describe all the interactions in the triple, then the all-pairwise interaction model will be a good fit as well. We can use it to our advantage here, since the only question we're trying to answer when looking at a tuple of size k is whether the presence of the tuple could have been explained by the interactions of lower order. The all-pairwise interaction model for a triple would be a bad fit (will underestimate the true count) only if the higher order interaction term is necessary, i.e. the entities in the triple are truly dependent on each other.

Combining the loglinear modeling to estimate counts with the Empirical Bayes ranking schema produces the Empirical Bayes Screening (EBS) algorithm.

3 EBS aka MGPS - Algorithm Description

Suppose the true counts of tuples of size k are drawn from a Poisson distribution: $n_{ij..k} \sim Pois(\lambda_{ij..k} e_{ij..k})$, where $e_{ij..k}$ are the estimates of the counts that we would have expected to see if we assumed that entities have co-occurred independently. Here, we assume that all counts come from different Poisson distributions, however as it was discussed above we need not assume that all λs are independent. In fact, we assume that all λs in turn came from a distribution. For simplicity, we assume that the underlying distribution was a mixture of gammas, hence the second name of the algorithm – MGPS (Mixture of Gammas - Poisson Screening). The assumption of this prior does not restrict us in any way, since a mixture of gammas is a sufficiently flexible model and is also a conjugate prior for a Poisson distribution, hence we can obtain a closed form formula for the posterior. In practice, we have discovered that for the majority of datasets, a single gamma prior was sufficient. The algorithm uses Empirical Bayes [Carlin and Louis, 2000] to estimate the parameters of the prior. The λs that we estimate from the posterior are in fact the smoothed version of the ratios of true counts to the independence estimates, but much more robust, since they are less influenced by the smaller sample size. The algorithm is described in detail in [DuMouchel and Pregibon, 2001;

DuMouchel, 1999], so we only give intuition and a brief outline of it here.

3.1 Estimates

Let N be the number of records in the dataset, n_i be the number of occurrences of entity i, n_{ij} number of occurrences of entities i and j together and $n_{ij...l}$ the number of occurrences of $i, j, ..., l$ accordingly. Then for pairs, the estimates are calculated based on the assumption that items i and j have occurred independently: $e_{ij} = N * p_i * p_j$, where $p_i = n_i/N$. For k-tuples where $k > 2$, the independence assumption is based on the loglinear model estimates. For example, for any given triple n_{ijk} we construct all 2-way marginal tables and use an iterative proportional fitting algorithm [Bishop *et al.*, 1977] to obtain the loglinear estimate e_{ijk}. The e_{ijk} will effectively reflect the estimate of the count we would have expected based on the all 2-way interaction model. Please note that since the size of the complete set of k-size tuples grows exponentially with k, we use the notion of support n^* (which maybe familiar to the association rules audience), so that we are only concerned with those k-tuples whose number of occurrences has exceeded the support.

3.2 Model

The data is modeled using hierarchical structure:

- $n_{ij..k}|\lambda_{ij..k} \sim Pois(\lambda_{ij..k}e_{ij..k})$
- $\lambda_{ij..k} \sim pGamma(\alpha_1, \beta_1) + (1-p)Gamma(\alpha_2, \beta_2)$, where p is the selection parameter between the two $Gamma$s. We estimate the parameters of the prior $\hat{\theta} = (\alpha_1, \beta_1, \alpha_2, \beta_2, p)$ by maximizing the likelihood of the product of the unconditional probability distributions for each n:
- $f(n) = \int Pois(n|\lambda e)\pi(\lambda|\hat{\theta})d\lambda$
- $\hat{\theta} = argmax_\theta \prod_{\forall n} f(n)$

For each (n, e) we then use the Bayes Rule to compute the corresponding distribution of λ: $\lambda \sim \frac{Pois(n|\lambda e)\pi(\lambda|\hat{\theta})}{f(n)}$ and then we calculate the geometric mean Λ for the purposes of scoring the given pair (n, e). The above procedure is not too computationally intensive. The computing time is listed in Section 4.

4 Results

The algorithm as described has been tested on a variety of datasets. However, since it is a ranking algorithm, the performance evaluation on the real world dataset maybe difficult, without knowing ground truths about the dataset in advance. Hence, we have created a simulation of the real world environment, but where the target was known.

The simulation was created as follows. Suppose there are 26 entities in the sample. Let's label them A-Z. Suppose there is a target triple of people, XYZ, who are collaborating and have had a few meetings that all three attended. The goal of the algorithm is to detect tuples such as XYZ even if the pairwise (XY, XZ and YZ) counts are unexceptional, and other triplets are more frequent, yet caused by promiscuity.

The dataset consists of 3450 meetings, where X, Y and Z have each occurred in at least 50 meetings. The number of meetings and in general, most of the parameters of the generative model had been assigned arbitrarily.

We had set the lower bound on the presence of X,Y and Z since we did not want their mere occurrence in the dataset to look suspicious.

We assume the dataset is generated by a background in which there are many pairwise friendships among A through Z. There are 59 pairwise friendships that generated the dataset.

- There are two pairwise friendships involving X.
- There are three pairwise friendships involving Y.
- There are four pairwise friendships involving Z.
- There are fifty pairwise friendships, each between a randomly-selected pair of letters between A through W.

Meetings are generated stochastically. Each pairwise friendship generated 60 meetings. Each meeting generated by a pairwise friendship between α and β included α and β and three other people drawn independently from a distribution in which some letters are more promiscuous (more willing to show up to meetings) than others. We had also made sure that while X, Y and Z have occured quite often by themselves and all together, their pairs did not occur as often as of other entitities with similar rates of occurences (otherwise the triple would be perfectly explainable by all-2way interaction model.

Thus the only intentional triplet of collaborators in the model are XY and Z. The question is, can the XYZ triple be detect from the data in the presence of so much distraction?

The overall participation is depicted in Figure 4.

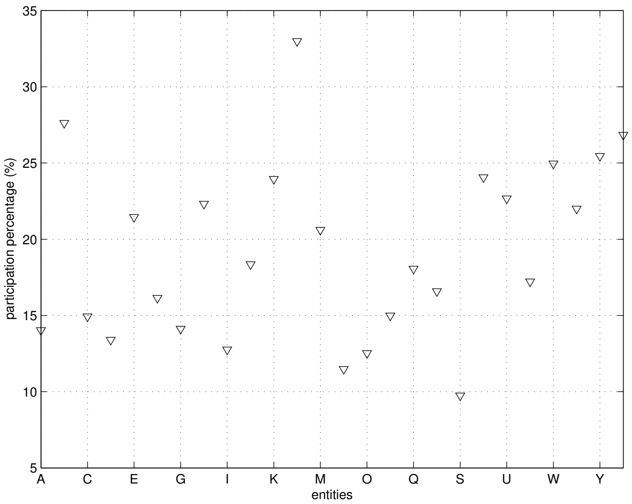

Figure 1: Participation distribution

From Figure 4 we can see that the participation of X, Y and Z is not less but even slightly higher than average. This, will undoubtedly thow off the algorithms based on complete independence assumption, since X, Y and Z are not 'rare' entities in the dataset.

triple	n
KLZ	74
BLZ	70
BLU	64
LTU	63
BFY	59

Table 1: Top 5 triples. Independence Assumption: either. Ranking method: n

We compared the performance of the following configurations of the algorithm:

- Ranking of triples based on the most frequent counts
- EBS based on the complete independence assumption ranked according to $r = n/e$
- EBS based on the complete independence assumption ranked according to Λ scores
- EBS based on the loglinear estimates, ranked according to r
- EBS based on the loglinear estimates, ranked according to Λ scores

The support n^* was set to 2. There were a total of 2560 triples that have exceeded n^*.

Based on the most frequent counts, the following top 5 triples were selected:

In fact, triple XYZ was ranked 125 having occurred 32 times.

When running the EBS algorithm assuming complete independence of items occurring in triples, the following rankings were obtained:

a) based on r

triple	n	r	Λ
CNR	18	1.84	1.41
GNR	17	1.83	1.40
BDX	50	1.78	1.61
MSW	30	1.74	1.48
MQS	21	12.5	1.68

Table 2: Top 5 triples. Independence Assumption: complete independence. Ranking method: r

Triple XYZ was ranked 793 having a ratio of 0.62. As expected the independence assumption has failed in this case, due to the high participation of all the members of the triple XYZ separately.

b) based on Λ

Based on Λ, the triple XYZ was ranked number 737. It is slightly better then the scoring produced by looking at ratios of counts to estimates, but is clearly not appropriate. The complete independence model is a very bad fit to our data. It is too simplistic to capture the complicated relations in the data, consequently even the Bayesian smoothing seems to fail provided with invalid estimates.

When we examine the results provided in Table 4 produced by the algorithm assuming the all 2 way interaction loglinear model, the ranking based on the simple ratio r does significantly better then the ranking based on the independence assumption. The triple XYZ is ranked number 17. However it is clearly far from optimal.

triple	n	r	Λ
BDX	50	1.78	1.61
MSW	30	1.74	1.48
DWX	41	1.62	1.45
CNR	18	1.84	1.41
BFY	59	1.51	1.40

Table 3: Top 5 triples. Independence Assumption: complete independence. Ranking method: Λ

triple	n	r	Λ
CDO	5	2.70	1.43
DGI	4	2.61	1.33
DEG	6	2.52	1.46
GSV	8	2.19	1.48
ADR	5	2.11	1.31

Table 4: Top 5 triples. Independence Assumption: all two-way interaction model. Ranking method: r

The full implementation algorithm taking advantage of the Bayesian shrinkage technique produces the desired output. The triple XYZ is ranked number 1 as can be seen in Table 5.

triple	n	r	Λ
XYZ	32	1.83	1.65
CNR	18	1.95	1.61
BGR	26	1.68	1.50
MNS	11	1.97	1.49
AFL	15	1.80	1.48

Table 5: Top 5 triples. Independence Assumption: all two-way interaction model. Ranking method: Λ

In fact, this was a hard case for our algorithm to detect since there were quite a few pairs XY, YZ, XZ making it look like the triples could be explained using the all two-way interaction model, but the model was found to be not a good fit, hence the algorithm has produced the desired output.

We can notice that the top choices of triples produced by various ranking techniques do not intersect. This can be explained by the complexity of the underlying generative model. Indeed if the naive assumption of complete independence finds the top choices to consist of entities that have a low probability of occurring the dataset, it is going to be highly surprised by random occurrences, whereas the complexity of the all two-way interaction independence assumption allows the algorithm to capture the complexity of the interactions in the dataset.

In this simulated dataset we have constrained ourselves to identifying a triple. A rather simplified task was chosen in order to keep the logistics of the simulation intuitive. However, the algorithm is capable of searching for k-tuples, where k is a reasonable number limited only by the power of the dataset. k would be unreasonable if it was so large that there are only a tiny number of k-tuples in the dataset that have occured more than once. Then it would not be necessary to fit a complicated model.

Table 6 shows the relative time performance based on several real world dataset. All tests have been performed on a Pentium 4 CPU 1.80GHz processor. The datasets were as follows:

1. *Citeseer* – a collection of co-publication links from the Citeseer online library. The dataset was slightly noisy due to the aliasing problem introduced by the encoding of the authors: first letter of the first name + last name.
2. *Drinks* – series of popular drink recipes found on various websites. Each drink was considered a link and the ingredients were entities in our schema.
3. *RI* – set of links of types: co-publication, common research interests, advisor/advisee collected from the Robotics Institute CMU web pages.
4. *Terrorism* – set of links created by a human who manually extracted information from news articles related to terrorism given a set of keywords and subjectively linked events and entities.

Table 6 shows the computational feasibility of our approach. Currently, the biggest bottleneck is in calculating counts for all tuples up to size k. This issue is going to be addressed in the future.

5 Discussion

5.1 Independence assumption

It can be seen from the algorithm that the smoothed scoring function greatly depends on the estimates of our counts that we obtain based on our independence assumption. As was shown in the toy example in the introduction, the naive assumption of complete independence between entities is not appropriate. For each k-tuple there are several loglinear models that could be a good fit to the data. We base the choice of our "all $k - 1$ way interaction model" on the fact that if it is not a good fit to the data, the best model would be the saturated one, i.e. the model that involves an interaction of all k entities. However, there is a slight complication due to the increasing sparsity of the data as k grows that is intrinsic to the problem. As in the terrorism dataset example, the number of all possible triples could have been $\binom{}{n3}$, but the number of triples that have co-occurred at least twice was 245 which is less than 1% of all possible triples! When using the loglinear models we have to be careful not to have too many sampling zeros, otherwise the estimates that we will obtain will be de-generate. There is a large body of research that tackles the problem of sampling and marginal zeros, for example [Agresti, 1990; Bishop *et al.*, 1977; Fienberg, 1980]. One way to deal with zeros is to add a very small value to the zero counts, as described in [Agresti, 1990], however that approach has its own disadvantages mentioned in [Fienberg, 1980]. Another possibility is to use the lower order loglinear model, for example all pairwise interaction even for quadruples, i.e. instead of using $u_{123} + u_{124} + u_{234} + u_{134}$ we could use $u_{12} + u_{13} + u_{14} + u_{23} + u_{24} + u_{34}$ as the independence model to obtain the estimate of the true count. Even if it does not seem to be the best choice from the real world perspective, since we are using it for smoothing algorithm, a consistent underestimated but statistically supported approximation to the estimates may be a better choice then a more complicated model with smaller degrees of freedom due to the sparseness. This issue remains contraversial in the statistical world and we need to conduct more experiments to make an empirical judgment about which technique works best based on different types of data.

5.2 Choosing number of gammas in the prior

In general we expect most people to be innocent so that their meetings with suspects could be attributed to randomness of people meeting others in the world. For example, many people go to airports and just by chance that could have been at the airport at the exact same time as some of the suspects, but we would not want to raise an alarm in such cases. If innocent people have co-occurred with suspected people due to the randomness in co-occurrences, it would be easily explained by the independence assumption. Thus, majority of the estimates of counts of co-occurrences should be pretty close to the true counts or scaled down by some factor that stays approximately constant throughout the set of count for the tuple of the interest size k. So, we expect our ratios $r = n/e$ to be roughly $c = const \approx 1$ in the majority of cases in the dataset. If that is the case, we do not want to use a mixture of gammas model, since that model will be prone to absorbing noise that might contain important information. The single gamma prior in that case is quite enough. However, there may occur situations when the ratios are not centered around any particular mean value, and then we do want to use the mixture of gammas for higher flexibility. The use of more than one gamma however causes a computational slow down, since we have to perform the maximum likelihood search over the space of 5 parameters. The datasets that the algorithm has been applied to, did not require the use of the mixture of gammas, but it is possible to use various computational tricks to speed up the search if needed. That would have to be looked into if the algorithm is to be applied to dense datasets with bi-model or a more complex distribution.

6 Extensions

There are several fields, in particular in the area of text mining, where this algorithm could contribute. For example, it could be used to combine labeled and unlabeled data. Suppose we know words A and B and we would like to assign labels to words D, E and F. We could count the number of occurrences of each of the labeled words with unlabeled ones looking at tuples of up to size 4 and then based on the ranking that our algorithm produces we could decide the labels of which known words fit the unknown words best, where we should put higher weight on the higher order interaction. In other words, if AD and AE were ranked higher then BD or BE, but BDE was ranked higher than ADE, then we should assign label B to words D and E contradictory to the predictions based on the pairwise interactions.

dataset	top quadruple	size (links x entities, filesize)	# 4tuples	comp freq (sec)	comp stats (sec)
Citeseer	{a_mehra j_rexford k_shin w_feng}	109099x104801 8,395KB	33,243	2046	56
Drinks	{ice orange_juice pineapple_juice rum}	5409x135 266KB	15,928	2	9
RI	{b_nagy d_wettergreen g_podnar j_bares}	1473x1329 88KB	27,549	1	5
Terrorism	{abu_mussa_group dflp pflp-gc syria}	5998x3743 310KB	222	12	1

Table 6: Relative speed performance based on several real world datasets

Another application would be word-sense disambiguation. Suppose we want to distinguish two types of *plant* sentences, where one is about flora and the other is about the factory. There are words that could occur in both types of sentences with relatively low frequencies, such as {*air, pollution, green*}, etc that would be considered good indicators of sentences of general topic *plant* but would fail in distinguishing between the two meanings of *plant*. In fact, when seeing a pair {*green, plant*}, we might classify the sentence to be of type *flora*, when in fact there was a word *paint* that we overlooked and then the combination {*plant, paint, green*} would indicate a preference toward the *factory* type, so clearly in this case the 3-tuple would give a more accurate idea of the meaning of the word *plant* in the given sentence than a simple pairwise independence assumption, thus it would be of great benefit to use our algorithm in this case.

7 Related Work

In spite of the age of the domain of Social Network Analysis and the recent increased attention to link analysis in various fields, there is an even larger diversity of approaches and questions that are being answered. A large branch of link analysis has approached problems in the domain using graph theory. Most of those algorithms, however, are either based on the assumption of existence of the underlying graph as described in [Wasserman and Faust, 1994] or search for the most likely graph structure based on the weighted counts of co-occurrences [Kubica *et al.*, 2003; Newman, 2001; Kautz *et al.*, 1997]. Inferring a structure of the underlying graph is different from the problem we are trying to address here. We are not trying to answer what is the most likely structure of the network that has lead to particular co-occurrences, but rather what are the most significant k-tuples that could not be explained by the lower relations.

There are quite a few probabilistic approaches in the field as well [Cohn and Hofmann, 2001; Friedman *et al.*, 1999; Kubica *et al.*, 2002]. [Cohn and Hofmann, 2001] have similarly assumed a mixture of sources for document generation, [Kubica *et al.*, 2002] has assumed a single source with noise added. Those methodologies could be regarded as a mixture or single distribution prior. However, in our approach we make use of hierarchical modeling, which allows us to assume that each count has come from its own Poisson distribution with mean λe, making the model significantly more flexible.

There has been also research involving loglinear models to infer the 'popularity' and the 'expansiveness' of the entities or 'actors' when looking at social networks [Wasserman and Faust, 1994]. However, the authors have stipulated that the problem of looking at interactions in k-tuples where $k > 3$ is too complex. They have also been looking at very small datasets.

One of the main advantages of the EBS algorithm in comparison with the majority of tools described in the link analysis literature is the ability to look beyond complete or pairwise independence. Those methods would fail in cases when the target relations in the dataset are of higher order, as was shown in the analysis of the simulated data.

Many questions similar to those we are trying to answer come from the association rules literature. Several papers such as [Silverstein *et al.*, 1998; Aggarwal and Yu, 1998] describe ranking methodologies based on the comparison of true to estimated counts. [Silverstein *et al.*, 1998] had used a more complex approach then the traditional support-confidence framework, but as described in [DuMouchel and Pregibon, 2001], there was a problem with the degrees of freedom in their model. The methodology of [Aggarwal and Yu, 1998] will be prone to over-estimation of 'interestingness' when dealing with small samples sizes.

8 Conclusion

In this paper we had presented a statistical approach to answering questions arising in the domain of link analysis such as 'what are the most *interesting* k co-occurrences of entities in the dataset'. The *interestingness* in this case is measured by the 'surprise' of seeing the true count given the estimate based on the independence assumption. There are several advantages to using this algorithm over naive estimations. First of all we are able to capture complex relationships among entities due to the loglinear model fitting when estimating the true counts. Second, the Bayesian shrinkage makes the methodology more robust when dealing with samples of various sizes. We have shown the strength of the approach on a simulated dataset, when the simpler assumptions had failed. We have also discussed possible usage of this algorithm in the domain of combining labeled and unlabeled data and for the word-sense disambiguation problem.

Acknowledgments

We would like to thank Stephen Fienberg (Statistics, CMU) and Ricardo Silva (CALD, CMU) for their insightful comments.

References

[Aggarwal and Yu, 1998] C. Aggarwal and P. Yu. A new framework for item set generation. In *Proceedings of*

ACM-PODS Symposium on Principles of Database Systems, pages 18–24, 1998.

[Agresti, 1990] A. Agresti. *Categorical Data Analysis*. John Wiley and Sons, New York, NY, 1990.

[Bishop *et al.*, 1977] Y. Bishop, S. Fienberg, and P. Holland. *Discrete Multivariate Analysis: Theory and Practice*. MIT Press, 1977.

[Carlin and Louis, 2000] B. Carlin and T. Louis. *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman and Hall/CRC, 2nd edition, 2000.

[Cohn and Hofmann, 2001] D. Cohn and T. Hofmann. The missing link - a probabilistic model of document content and hypertext connectivity. *Neural Information Processing System*, 13, 2001.

[DuMouchel and Pregibon, 2001] B. DuMouchel and D. Pregibon. Empirical bayes screening for multi-item associations. In *Proceedings of the Seventh ACM SIGKDD Internation Conference on Knowledge Discovery and Data Mining*, pages 67–76, 2001.

[DuMouchel, 1999] B. DuMouchel. Bayesian data mining in large frequency tables, with an application to the fda spontaneous reporting system (with dicussion). *The American Statistician*, 53:177–202, 1999.

[Fienberg, 1980] S. Fienberg. *The Analysis of Cross-Classified Categorical Data*. MIT Press, 2nd edition, 1980.

[Friedman *et al.*, 1999] N. Friedman, L. Getoor, D. Koller, and A. Pfeffer. Learning probabilistic relational models. In *IJCAI*, pages 1300–1309, 1999.

[Kautz *et al.*, 1997] H. Kautz, B. Selman, and M. Shah. The hidden web. *AI Magazine*, 1997.

[Kubica *et al.*, 2002] J. Kubica, A. Moore, J. Schneider, and Y. Yang. Stochastic link and group detection. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence*, pages 798–804. AAAI Press/MIT Press, July 2002.

[Kubica *et al.*, 2003] J. Kubica, A. Moore, D. Cohn, and J. Schneider. Finding underlying connections: A fast graph-based method for link analysis and collaboration queries. Submitted to ICML, 2003.

[Newman, 2001] M. Newman. Who is the best connected scientist? a study of scientific coauthorship networks. *Phys Rev*, 2001.

[Silverstein *et al.*, 1998] C. Silverstein, S. Brin, and R. Motwani. Beyond market baskets: generalizing association rules to dependence rules. In *Data Mining and Knowledge Discovery*, pages 39–68, 1998.

[Wasserman and Faust, 1994] S. Wasserman and K. Faust. *Social Network Analysis : Methods and Applications*. Cambridge Univ Press, 1994.