Determining Patient Similarity in Medical Social Networks

Sebastian Klenk, Jürgen Dippon, Peter Fritz, and Gunther Heidemann

Stuttgart University
Intelligent Systems Group
Universitätsstrasse 38, 70569 Stuttgart, Germany
ais@vis.uni-stuttgart.de

Abstract. In social networks the primary concern of people is to find others who share similar interests. For medical systems this means finding people who have similar symptoms or comparable diseases. Here a simple matching of variables would lead to a very small number of identical cases and determining similarity would usually fail due to the categorical nature of most factors. In particular, such problems arise for cancer patients. We have developed a system that is capable of determining similarity in terms of the survival time distribution. By a similarity based search our approach allows to determine related patients. Thus recommendations for contacts of interest become possible. We will present the theoretical foundation as well as a use case scenario with an existing data mining software.

1 Introduction

Finding "patients like me" is a big issue for people suffering from severe illness. Today, this problem is addressed by the medical social network with identical name [1], and by organizations such as the german ACHSE[2] or the european Eurordis[3], which represent the common interests of patients and have brought together people with similar diseases successfully for several years now.

The goal of most medical social web sites is to provide a forum and a more direct way for patients to exchange thoughts, feelings, and experiences. Therefore the search for other people with a similar disease history and similar symptoms is crucial. For this purpose, patient profiles are presented which share a large number of similarities, just like in other social networks. However, defining such similarities for patient profiles is significantly more difficult than for other types of social networks. Different aspects of a disease have to be weighted differently, so a simple matching of factors is insufficient.

We have developed a similarity measure for cancer patients which calculates influence values for factor levels and thereby facilitates a soft matching. This

[1] PatientsLikeMe is a social networking health site with over 40,000 Members http://www.patientslikeme.com

[2] The German Alliance for Rare Chronic Diseases http://www.achse-online.de

[3] Eurordis – Rare Diseases Europe http://www.eurordis.org

means that different aspects are also weighted differently. For example, the fact that two cancer patients have developed metastasis is weighted much higher than similar age. This leads to a domain specific matching and provides better recommendations on who might have had similar experiences or who might have knowledge a user can benefit from. Finding relationships of this kind is the very basis of social media.

2 Related work

An important part of social networking research [17] is on *recommender systems* [5, 10, 8, 16]. These are systems that recommend certain items to the user, usually products, but also people one might want to know. As this is particularly interesting for e-commerce applications, most research is on suggesting new products.

For recommending people, there are two common approaches: (*i*) Content based recommendation which uses the information the user enters into the social network application, whereas (*ii*) relationship based recommendation traces who are the friends of the users friends, which the user might want to meet. Chen et. al. [2] provide an overview on both fields and perform a comparative study. Their results are mostly in favor of the relationship based approach, whereas they argue that similarity in content is so far calculated by keyword matching, which is just not sufficient. An example for a relationship based method is the work of Lin et. al. [12, 7], who deal with the problem of matching people in the context of searching for experts. They combine a graph based approach with a matching of search terms with profile terms which yields good results. But in the case of medical data an approach of this kind would lead to insignificant results because term matching does not reflect the true difference of the underling objects. Here more detailed domain knowledge is be required to determine term weightings. As stated by both Felfering et. al. [8] and Volinsky [16], deep domain knowledge is so far not used excessively in recommender systems.

It is obvious that content based recommendation is, at least in principle, superior to relationship based recommendation, as it would allow to explore the entire network rather than just the subset a user is connected to. We therefore aim at improving content based recommendation by making an interpretation of the given content feasible.

Another important aspect of a weighted content based approach is security. Such a system is less likely to be subject to fraud or spamming as described by Mobasher et. al [13].

Apart from recommender systems, distance learning has a long history in the area of case based reasoning [14]. Learning distance measures facilitates a context sensitive estimation of similar cases. Arshadi and Jurisica [1] employ logistic regression to estimate a distance measure which gives relevance to certain aspects of the data. The method we describe here differs from the one proposed by Arshadi and Jurisica, as it allows for continuous dependent data which can even be censored, a feature that is crucial for medical data.

The distance measure we are using here is based on an idea proposed in [6], which has been extended and implemented in the medical data mining system OCDM [11]. In the present paper we present a new application of this idea in the context of medical social networks.

3 Similarity for patient data

Measuring the similarity of natural continuous data items is very much straight forward. Every data dimension has the same weight and differences between dimensions can be interpreted in a very intuitive way. For categorical and artificial data, as is the case for patient data, differences in variables are anything else but intuitive and the weighting varies with each dimension. Formally speaking for two data items x and y a distance looks as follows:

$$d(x, y) = \sum_{k=1}^{n} \alpha_k d_k(x_k, y_k). \tag{1}$$

Here $\alpha = (\alpha_i)_{i=1...n}$ is a weighting term that is assigned to each dimension and corresponds to its influence on the similarity. When working with lung cancer patient data for example it makes a huge difference whether the patient smokes or not but the area he or she lives in is of minor importance. Therefore similarities for smoker (yes or no) should have higher α values than for similarity in zip code.

Besides the weighting factor there is also the functions d_k which could be the absolute, the squared or the binary distance

$$d_k(x, y) = 1 \quad \text{if} \quad x = y \quad \text{else} \quad d(x, y) = 0$$

depending on the dimension k.

Determining a suitable weighting is essential to finding a good similarity measure. Therefore it is necessary to have a method at hands to calculate such a weighting. The central idea to the similarity measure learning approach we have taken, is to have a linear relation between a number of independent and one dependent variable that can be estimated and used as a weighting scheme. An ideal candidate to estimate such a scheme is the logistic regression [9]. This is a supervised learning scheme that, based on training data, estimates the influence a given set of independent variables has on a dependent variable.

Formally it calculates the probability of a variable G having a certain value g given the information contained in all the other variables $X = x$

$$P(G = g|X = x) = \frac{\exp(\beta_g^T x)}{1 + \sum\limits_{g' \in G} exp(\beta_{g'}^T x)}. \tag{2}$$

This formula gives us the influence each element x_i of x has on the outcome g of G. Here the weight vector β represents this information. Equation (2) can be used to model this influence for discrete data, for continuous and censored dependent variables, Cox has developed a method to calculate β [3]. The central thought of his work is that the function $h(t|x)$ can be described as

$$h(t|x) = h_0(t) \cdot P(h = h_0|X = x), \tag{3}$$

where $h_0(t)$ is unknown. This leads to

$$h_0(t) \cdot \exp(\beta^T x). \tag{4}$$

What is actually estimated in (3) and (4) is the distribution function of the survival times. It is based on an unknown baseline hazard function that determines the risk of a patient at a certain moment. The formula in (3) is known as Cox proportional hazard regression or just Cox regression and is mostly used in survival analysis [4, 15]. The actual estimation of these parameters takes place with a Newton-Raphson based method. Therefore the partial log likelihood function (for the parameter β over a training set) is maximized.

Given the influence information β out of (3), it is easy to develop a distance measure that is sensitive to the relevant aspects of the data concerning the variable for which the estimator was trained.

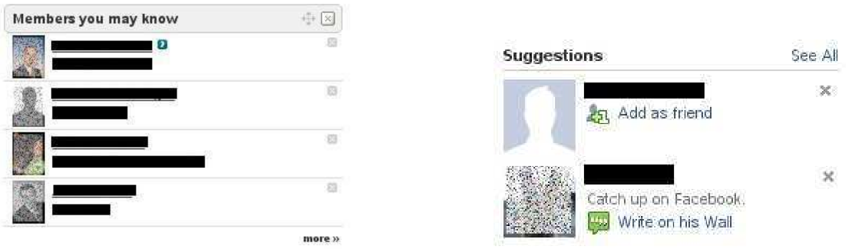


Fig. 1. The recommendation of people in other social networks (on the left side Xing and on the right side Facebook)

3.1 Patient recommendations by regression estimation

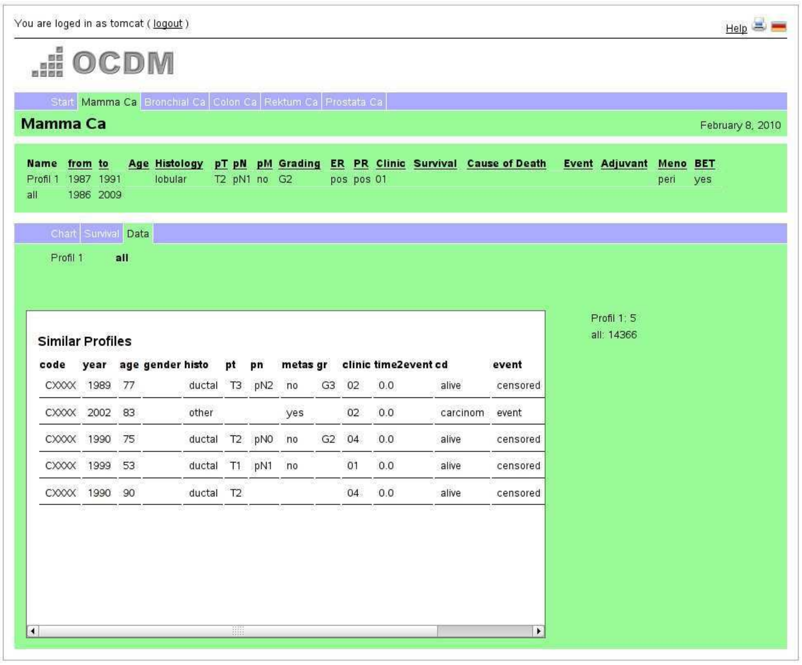
Recommendations of other people in a social network is a central theme of social applications (see also Figure 3 for examples). In the above section we have described how regression estimation can lead to a weighting of variables and thereby allow for the calculation of specific distance measures. Here we will describe how such a measure can be used to determine other people in the social network that one might want to know.

A social network application consist of a large database containing information on the people belonging to it. The information was entered by the people themselfs and may therefor contain only certain aspects of their profile. To determine other people with similar views it is necessary to calculate a distance measure as described above. Given a database with sample cases (the training data) one is able to estimate the weighting parameter β and apply it to a distance measure of the form:

$$d(x, y) = \sum_{k=1}^{n} \alpha_k d_k(x_k, y_k).$$

Here $\alpha = (\alpha_i)_{i=1\ldots n}$ with $\alpha_i = \exp^{\sigma \cdot \beta}$ and σ being a scaling factor to match the influence of the weighting on the distance measure. The measure itself could be the squared distance $d_k(x, \tilde{x}) = ||x - \tilde{x}||^2$ or simply the absolute distance. The scaling factor it self can be chosen to suite the needs of the recommendation, should the influence of the independent variables on the survival be weighted more heavily a value of $\sigma >> 1$ should be selected, in any other case $\sigma \leq 1$ is a good choice.

Now if this measure is applied to all people in the database one obtains a partially ordered list where the first few profiles can be used as recommendations. To reduced computational load one could restrict the number of computations by only considering profiles that share a least amount of common fields.



You are loged in as tomcat (logout)

Mamma Ca

February 8, 2010

Name	from	to	Age	Histology	pT	pN	pM	Grading	ER	PR	Clinic	Survival	Cause of Death	Event	Adjuvant	Meno	BET
Profil 1	1987	1991		lobular	T2	pN1	no	G2	pos	pos	01					peri	yes
all	1986	2009															

Profil 1 **all**

Similar Profiles

code	year	age	gender	histo	pt	pn	metas	gr	clinic	time2event	cd	event
CXXXX	1989	77		ductal	T3	pN2	no	G3	02	0.0	alive	censored
CXXXX	2002	83		other			yes		02	0.0	carcinom	event
CXXXX	1990	75		ductal	T2	pN0	no	G2	04	0.0	alive	censored
CXXXX	1999	53		ductal	T1	pN1	no		01	0.0	alive	censored
CXXXX	1990	90		ductal	T2				04	0.0	alive	censored

Profil 1: 5
all: 14366

Fig. 2. The presentation of similar patients in the OCDM system

4 Implementation

We have implemented the similarity distance measure in our data mining software OCDM [11], where similar patients are found for a given patient profile. This system, although intended for physicians, recommends similar profiles for a further study. For a patient an identical approach could lead to a recommender system as described above. In this section we are describing technical details about the similarity search. We will thereby concentrate on rather generic technical aspects, further details about the actual implementation of the similarity search can be found in [11]. As basis for the developed system serves a PostgreSQL Database Server and a Java-Tomcat Servlet-Engine. As performance is a critical aspect of the software and much calculation has to be done during the estimation of the distance measure (on one hand the calculation of the weights and on the other the similarity calculation when recommending other profiles) we didn't follow a strict layer separation. Some tasks that involved extensive data processing were developed as stored procedures that run inside the database process. Most of the heavy-load calculation was thereby separated from the middleware and the GUI. As some of the calculation procedures are needed in the stored procedures and in the business logic we implemented these as Java classes such that they could be used in PL/Java code in the database as well as plain Java objects in the application server. We did some experiments with the similarity based distance we have developed and thereby achieved results comparable to that of common SQL queries. We measured the time it took for the database server to return results. For a data set of roughly 15.000 cases the database returned the select data on average after 10 milliseconds whereas the similarity based search took 35 milliseconds. These results can be placed in context when looking at the time it takes to process a simple SELECT statement with a function term (adding a constant to a column value) or a SELECT statement with an aggregate (calculating the average of a column value). The results are summarized in Table 1.

Query Type	mean	std-err.
Simple SELECT	9.29	6.28
SELECT with function term	24.81	8.29
SELECT with aggregate	95.60	11.31
Similarity Search	35.09	10.50

Table 1. Time until results are returned in milliseconds

5 Discussion

We have presented a domain specific distance measure for medical social networks. It is not intended to be generally applicable to the broad audience of medical social networks, rather, it allows certain groups of patients to obtain better recommendations. If it is known that a user suffers, e.g., from a certain cancer type, search for other network members is focused and directed by criteria specific to this disease. The weighting in the actually calculated distance measure (1) can be easily adapted to a particular user group. Another important aspect of the above described distance measure is that it is solely focused on the survival time and does not include other possibly relevant aspects such as regional proximity or corresponding interests. In our experience, this restriction led to the best results. However, the restriction can be easily removed to include combinations of different weighting schemes. For two given weighting vectors α^1 and α^2 it is easy to combine them to a new weighting scheme α^* by just summing up corresponding normalized elements

$$\alpha_i^* = \frac{1}{2 \cdot ||\alpha^1||}\alpha_i^1 + \frac{1}{2 \cdot ||\alpha^2||}\alpha_i^2.$$

Data coding and treatment of missing values are important issues, because not every user will conform to standardized nomenclature to describe his or her disease, and likewise, many users will not present all their information in a social network. Both data coding and missing values have significant influence on distance estimation. Missing values can already be handled by the parameter estimation procedure and the distance measure itself as well. So the remaining problem is the lack of a formal notation. This, of course, could dramatically decrease the efficiency of the training process (if it is based on the data in the network). However, social networks have grown at such pace in the recent years that it is still highly likely to find a sufficient number of "good" training samples, even if data with unclear values have to be omitted. When it comes to proximity calculation, informal and varying notation could be handled in such a way that only those variable values that match certain criteria are considered for calculation, while all others are treated as missing values.

6 Conclusion

We have presented a method to calculate similarities of patient profiles for recommending people to other members in a social network. As connecting to other people is the central aspect of medical social networks, a subject specific similarity search can increase the performance of recommendations and thereby increase the usefulness of the social network application dramatically. In addition to presenting the theoretical foundation we also have given insight into some implementation details as well as performance measures. These show comparable results to more complex SQL queries and can serve as a guideline when implementing a similar approach in a real world application. As the method we have presented is highly subject specific, i.e., dependent on the estimation of survival time data, it might be interesting to see further research on other medical data that might be less dependent on a time to event. Further the incorporation of social graph information seems to be promising.

References

1. N. Arshadi and I. Jurisica. Data mining for case-based reasoning in high-dimensional biological domains. *Knowledge and Data Engineering, IEEE Transactions on*, 17(8):1127–1137, Aug. 2005.
2. Jilin Chen, Werner Geyer, Casey Dugan, Michael Muller, and Ido Guy. Make new friends, but keep the old: recommending people on social networking sites. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, pages 201–210, New York, NY, USA, 2009. ACM.
3. D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society Series B (Methodological)*, 34(3):187–220, 1972.
4. David R. Cox and E. J. Snell. *Analysis of binary data.* Monographs on statistics and applied probability ; 32. Chapman and Hall, London, 2. ed. edition, 1989.
5. M. Deshpande and G. Karypis. Item-Based Top-N Recommendation Algorithms. *ACM Transactions on Information Systems*, 22(1):143–177, January 2004.
6. J. Dippon, P. Fritz, and M. Kohler. A statistical approach to case based reasoning, with application to breast cancer data. *Comput. Stat. Data Anal.*, 40(3):579–602, 2002.
7. Kate Ehrlich, Ching-Yung Lin, and Vicky Griffiths-Fisher. Searching for experts in the enterprise: combining text and social network analysis. In *GROUP '07: Proceedings of the 2007 international ACM conference on Supporting group work*, pages 117–126, New York, NY, USA, 2007. ACM.
8. Alexander Felfernig, Gerhard Friedrich, and Lars Schmidt-Thieme. Guest editors' introduction: Recommender systems. *IEEE Intelligent Systems*, 22:18–21, 2007.
9. Trevor J. Hastie, Robert J. Tibshirani, and Jerome H. Friedman. *The elements of statistical learning.* Springer, corrected print. edition, 2002.
10. Przemysław Kazienko and Katarzyna Musiał. Recommendation framework for online social networks. In *Advances in Web Intelligence and Data Mining*, Studies in Computational Intelligence, chapter 12, pages 111–120. 2006.
11. S. Klenk, J. Dippon, P. Fritz, and G. Heidemann. Interactive survival analysis with the ocdm system: From development to application. *Information Systems Frontiers*, 2009.
12. Ching-Yung Lin, Kate Ehrlich, Vicky Griffiths-Fisher, and Christopher Desforges. Smallblue: People mining for expertise search. *IEEE MultiMedia*, 15(1):78–84, 2008.
13. Bamshad Mobasher, Robin Burke, Runa Bhaumik, and Chad Williams. Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Trans. Internet Technol.*, 7(4):23, 2007.
14. Petra Perner, editor. *Case-based reasoning on images and signals : with 30 tables.* Studies in computational intelligence ; 73. Springer, Berlin, 2008.
15. Steve Selvin. *Modern applied biostatistical methods using S-Plus.* Monographs in epidemiology and biostatistics ; 28. Oxford University Press, New York, 1998.
16. Chris Volinsky. Matrix factorization techniques for recommender systems. volume 42, pages 30–37, 2009.
17. A.C. Weaver and B.B. Morrison. Social networking. *Computer*, 41(2):97–100, Feb. 2008.