RETHINKING RECRUITMENT

LEARNING LATENT INTERESTS FOR BETTER JOB RECOMMENDATIONS

Anthony Chow
ac1408@stanford.edu

Chang Su
changsu@stanford.edu

Junji Ma
majunji@gmail.com

Advisors: Jure Leskovec, Andreas Weigend
Dataset provided by: BranchOut

Abstract

We address the challenges associated with the matching of a job-seeker to a recruiter based on their interests. This is not trivial due to the inherent disconnect between what a job-seeker puts on his resume and what a recruiter puts on his job description. In addition, job recommendation should be also based on a user's transferrable skills, so that he gets exposed to as many job opportunities as possible that is within his capabilities. This paper proposes a new method to model a user's latent interests to infer his relevant and transferrable skills.

We apply our model onto a real dataset provided by BranchOut, a job search platform on Facebook. We demonstrate that the model is able to scale with historical data for all users in BranchOut. Interesting concepts could also be learnt between a user's description of self and a job's requisition. We also show that this model allowed for a much richer set of recommendations, which is done based on a user's transferrable skills and not based on his expressed skills. Finally, we demonstrate that a user's online engagement is also a key component in signaling if a user will apply for a job.

1. Introduction

Hiring the perfect candidate for your job is non-trivial. It used to be where a recruiter merely scans a candidate for relevant skill sets. It is now not the case anymore. Hiring has become less of a one-sided search and more of a mutual matching procedure. We often find that the best hires come from candidates whose intrinsic motivations are aligned with the company. This goes beyond merely looking at alignment of skills, but also of character and aspirations.

However, the problem is that the typical models for matching algorithms are largely based on the paradigm of keywords matching.[1] While it has achieved some form of success, it has a huge flaw due to the presence self-expression and creativity. It is often the case where a user describes his skill-set and background in a way that is very different from what a job description requires, and pure keywords matching tend not to pick up inherent similarities.

An ability to recognize the similarities between the user and job descriptions has applications in recommendation systems not restricted to job-matching, but to many other content matching purposes. More importantly, inference of a user's interest and skills would also allow better recommendations based on transferable skills and not simply based on a user's listed skills. For example, a user who has held the job of a cashier before could be inferred to have skills of 'customer service' and ' multi-tasking' that could turn up a set of more interesting job opportunities for him.

In this paper, we explore and propose 2 methodologies respectively that could identify these inherent similarities. In the first methodology, we show that we were able to learn the concepts that exist between the space of the user and job descriptions. These concepts will act as a bridge that connects the different form of expression of similar ideas by the user and the job. For example, a concept 'Technology' could be the connection between a user's skill of 'database management' and a job looking for 'software engineers'. These concepts could also allow for recognition of latent interests and transferrable skillsets of the user. The second methodology is inspired by an existing paper [3], which will demonstrate a different take on the same problem.

We finally then demonstrated a set of experiments that validated our model and hypothesis.

2. Problem Formulation and Dataset

To do that, we would need a real dataset to work on. BranchOut presented itself as the perfect dataset for our purposes because of the following. BranchOut is a job search platform mounted on the Facebook platform. Each user now brings with him rich data in his online identity as he searches for a job on BranchOut. This data is multi-dimensional, consisting on what he says and what he does on the platform. These include his resume, tagged skills, search history as well as the plethora of engagements he can do on BranchOut's highly interactive interface. This rich set of features gives the user an identity that is able to describe him to a high degree of granularity. We thus feel that it holds the key to an improved recruiter-candidate matching algorithm.

We thus began with a preprocessing of BranchOut's dataset. For a given user and job, we want to consider the task of predicting if the user will apply for the job. Therefore, we first extracted the relevant parts of the dataset to our problem. Specifically, we extracted about 16000 user-job pairs where the user viewed the job and applied, and another randomly selected 16000 user-job pairs where the user has viewed the job but did not apply. The equal ratio was to allow no biasness in training of our model. Of this samples, we extracted the following information:

User	Job
Description: Resume, Skills and Query (Search) History	Description: Job Requisition
Online Behavior: *(Appendix 1)*	

Table 1: User and Job features

This formulation has an eventual goal in predicting whether a given user will apply to a given job. While this paper will present the complete solution, we will focus largely on the methodologies employed to meaningfully match the user and job descriptions.

3. Related Work

We looked into several past works done on such problems. We found that learning topics to replace keyword matching has been a topic that has been gaining attention in the past few years. Studies have shown that there is a huge problem with keyword mismatch, especially in the online advertising space. [2] In fact, a recent paper written by Chi *et al*, on learning relevance from heterogeneous social network was targeted at learning these concept spaces that are lower in dimension and could avoid the problem of keyword mismatches. [3] Also, there have been other papers that used the friendship graph, where they hypothesized that the friendship link can be used to propagate interests, which could augment the keyword-matching paradigm. [4] We feel that this could be useful for our situation as well, as there is definitely a certain kind of influence exerted by a user's friends on his decision to look for a certain job. However, there have also been papers that suggest that this may not necessary lead to an improvement in quality. [5]

In the next section, we will explore the model that was proposed by Chi *et al* [3] and adopting it to fit to our case. We will then also propose another model that could also lead to better results in topic extraction by decoupling the dimensionality reduction matrix for the two parties involved.

4. Our Proposed Model

Here we present the 2 models. Their respective benefits as well as the experimental results will be discussed separately. But before there, here are some definitions.

A user vector: $U \in \mathbb{R}^n$
A job vector: $J \in \mathbb{R}^n$

The two vectors are constructed via the following means. We first searched for the words with the highest support in the space of user description as well as the job description. To get the words, we did the following 3 steps. We first removed all stopwords, parsed all the words into a Part-Of-Speech extractor, and then lemmatized them. This will ensure that we do not include words that do contribute to meaning, as well as avoid double counting of words due to differences in their grammatical forms. We then selected the top 1500 words with highest support from each space and represented the user and job vectors as binary vectors of these 1500 words.

We also define the following sets:

$$T = set\ of\ (user, job)\ applied\ pairs$$
$$\bar{T} = set\ of\ (user, job)\ not\ applied\ pairs$$

Model 1: The M&N Matrices

The idea of this model is to introduce 2 matrices, a user-word mapping matrix N and a job-word mapping matrix M.

$$M \in \mathbb{R}^{n \times m}$$
$$N \in \mathbb{R}^{n \times m}$$

where m << n

For a user i and a job j, we can then reduce the dimension of the User and Job representation by the following inner products:

$$U_i' = U_i^T M$$
$$J_j' = J_j^T N$$

The intuition is that these matrices will transform the job and user representation into the same space of lower dimensionality, with each dimension representing a distinct topic or concept. With these matrices, the problem formulation is the following. We want to minimize the following objective function, subject to the constraint that distances between a user-job applied pair is smaller than distance between a user-job not applied pair:

$$\min \|M\|_2^2 + \|N\|_2^2 + \sum_{(U,J_{Apply})\in T} \sum_{(U,J_{Not\ Apply})\in \bar{T}} h\left(d\left(U,J_{Apply}\right) - d\left(U,J_{Not\ Apply}\right)\right)$$
$$ST\ d\left(U,J_{Apply}\right) < d\left(U,J_{Not\ Apply}\right)$$
$$where\ d(U,J) = \sum_i (U_i' J_l')^2$$
$$and\ h(x) = \begin{cases} 0 & x < 0 \\ x^2 & x > 0 \end{cases}$$

From there, we applied stochastic descend to solve for the results. The following is the stochastic descend equations:

$$\nabla m_{ij} = \frac{\partial f}{\partial m_{ij}} = m_{ij} + \gamma \sum_{(U,J_{Applied})\in T} \sum_{(U,J_{Not\ Applied})\in \bar{T}} \frac{\partial h}{\partial m_{ij}}$$
$$m_{ij} \leftarrow m_{ij} - \eta \nabla m_{ij}$$
$$\nabla n_{ij} = \frac{\partial f}{\partial n_{ij}} = n_{ij} + \gamma \sum_{(U,J_{Applied})\in T} \sum_{(U,J_{Not\ Applied})\in \bar{T}} \frac{\partial h}{\partial n_{ij}}$$
$$n_{ij} \leftarrow n_{ij} - \eta \nabla n_{ij}$$

$$where\ \frac{\partial h}{\partial m_{ij}} = \begin{cases} 2(U^a M(:,j) - J^a N(:,j))U_i - 2(U^b M(:,j) - J^b N(:,j))U_i & if\ d\left(U_{Applied}^a, J_{Applied}^a\right) > d(U_{Not\ Applied}^b, J_{Not\ Applied}^b) \\ 0 & otherwise \end{cases}$$
$$where\ \frac{\partial h}{\partial n_{ij}} = \begin{cases} -2(U^a M(:,j) - J^a N(:,j))J_i + 2(U^b M(:,j) - J^b N(:,j))J_i & if\ d\left(U_{Applied}^a, J_{Applied}^a\right) > d(U_{Not\ Applied}^b, J_{Not\ Applied}^b) \\ 0 & otherwise \end{cases}$$

$$where\ A(:,j)\ refers\ to\ the\ jth\ column\ of\ matrix\ A$$
$$and\ U^a, J^a \in T\ and\ U^b, J^b \in \bar{T}$$

We then utilized batch descend method to run the gradient descend.

Model 2: The W Logistic Regression

This is the model that was inspired from Chi *et al*. [3]

We first define a square weight matrix W

$$W \in \mathbb{R}^{n\times n}$$

The definition for the user and job vectors are represented slightly differently in this case. Instead of obtaining top support words from the separate space of user and job descriptions, we obtain top support words in the combined space of user and job descriptions.

Associate a score with each user vector U and job vector J:

$$S = UWJ^T$$

We used a logistic function to define probability if a given User will apply to a given Job

$$P\left(U, J_{apply}\right) = \frac{e^{\beta s}}{1 + e^{\beta s}}$$

We then define a loss function:

$$L(T, \bar{T}) = \prod_{(U, J_{Apply}) \in T} P(U, J) \prod_{(U, J_{Not\ Apply}) \in \bar{T}} (1 - P(U, J))$$

With that, we aimed to maximize the logarithm of it to obtain the weights of our W matrix.

$$\max log\left(L(T, \bar{T})\right) = \sum_{(U, J_{Apply}) \in T} log\left(P(U, J)\right) + \sum_{(U, J_{Not\ Apply}) \in \bar{T}} (1 - log\left(P(U, J)\right))$$

We then used stochastic gradient descend to work out the results. The stochastic descend equations are as follows:

$$\nabla w_{ij} = \frac{\partial f}{\partial w_{ij}} = w_{ij} + \sum\nolimits_{(U, J_{Applied}) \in T} \frac{\beta}{1 + e^{\beta s}} U_i J_i - \sum\nolimits_{(U, J_{Not\ Applied}) \in \bar{T}} \frac{\beta e^{\beta s}}{1 + e^{\beta s}} U_i J_i$$

$$w_{ij} \leftarrow w_{ij} - \eta \nabla w_{ij}$$

We then utilized batch descend method to run the gradient descend.

Baseline Calculation 1: TF-IDF Similarity

Define a vector that represents the TF-IDF weights of each word in the word space:

$$F \in \mathbb{R}^n$$

Each element f_i is defined as follows:

$$f_i = tf_i \times log \frac{total\ num\ of\ documents}{total\ num\ of\ documents\ containing\ element\ i}$$

And:

$$tf_i = num\ of\ element\ i\ in\ document$$

We use the cosine similarity between the user description u and job description j:

$$sim(U, J) = \frac{F_J F_U^T}{\|F_U\| \|F_J\|}$$

Baseline Calculation 2: SVD dimensionality reduction

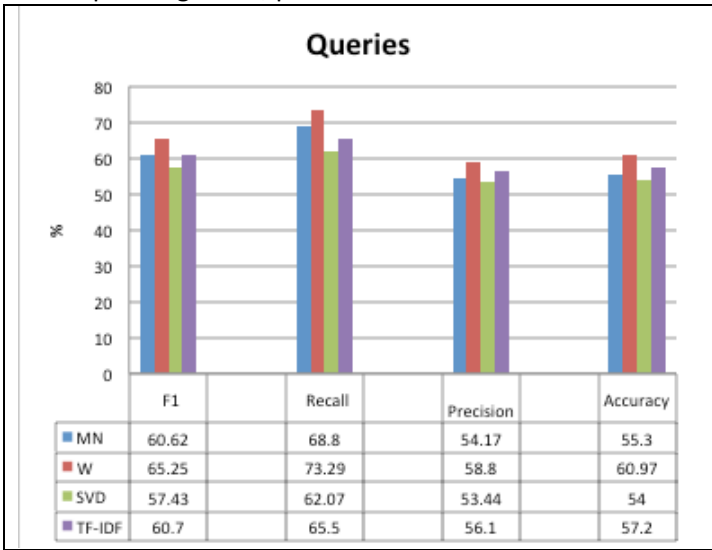
Concatenate all the user vectors into 1 User Matrix $\mathcal{U}$, with each row representing 1 user vector. Then concatenate all job vectors into 1 Job Matrix $\mathcal{J}$, with each row representing 1 job vector, such that $row\ i$ of User Matrix corresponds to either a user-job applied pair or user-job not applied pair with $row\ i$ of Job Matrix.

$$\mathcal{U} = USV^T \text{ and } \mathcal{J} = U'S'V'^T$$

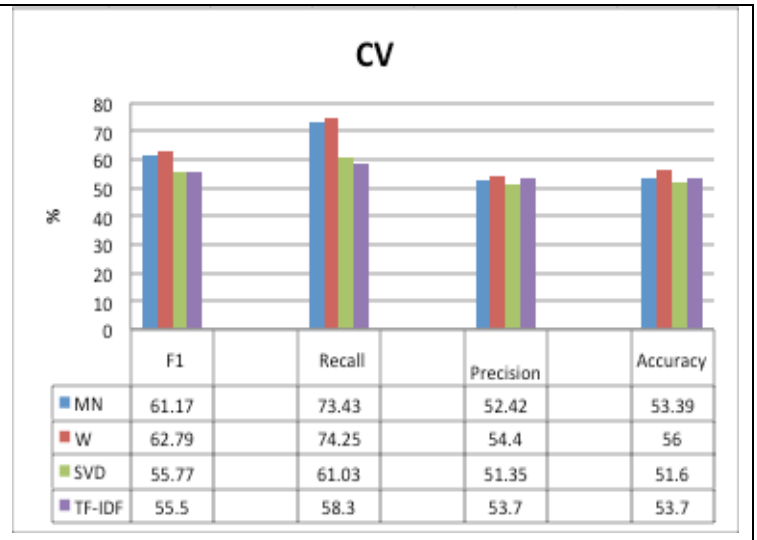
We then selected the 20 dominant eigenvalues in the S matrices, and reduced the dimensions of User Matrix $\mathcal{U}$ and Job Matrix $\mathcal{J}$ to 20 dimensions. The Euclidean or cosine distance between a given user and job pair can then be calculated to get a measure of the distance between them.

5. Experiments With Real Data

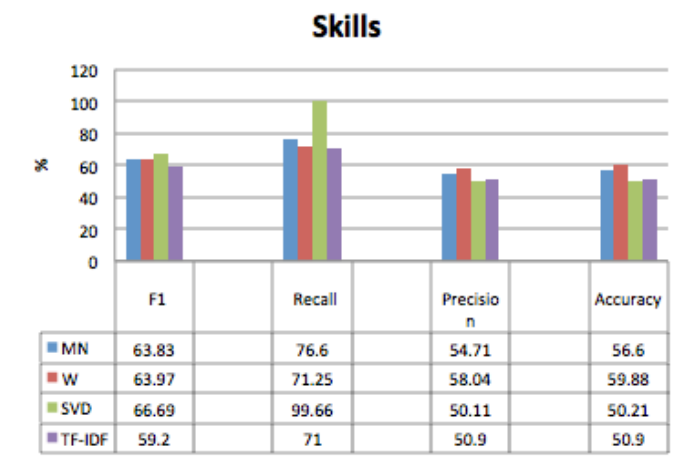
Here are our results comparing the 2 methods with our baseline methods. The evaluations were done as follows. We calculated the scores for the test data (55% of dataset) with the trained matrices for M&N Topics as well as W Regression. For TF-IDF and SVD, we worked based on the entire dataset. For each method, we then sorted the scores in ascending order, and looked for the partition that gave the best accuracy in terms of prediction. We then looked for the corresponding recall, precision and associated F1 score.



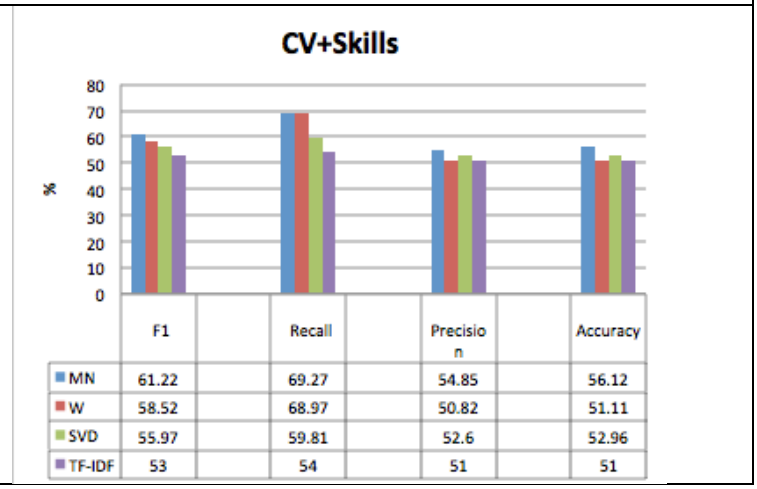
	F1	Recall	Precision	Accuracy
MN	60.62	68.8	54.17	55.3
W	65.25	73.29	58.8	60.97
SVD	57.43	62.07	53.44	54
TF-IDF	60.7	65.5	56.1	57.2



	F1	Recall	Precision	Accuracy
MN	61.17	73.43	52.42	53.39
W	62.79	74.25	54.4	56
SVD	55.77	61.03	51.35	51.6
TF-IDF	55.5	58.3	53.7	53.7



	F1	Recall	Precision	Accuracy
MN	63.83	76.6	54.71	56.6
W	63.97	71.25	58.04	59.88
SVD	66.69	99.66	50.11	50.21
TF-IDF	59.2	71	50.9	50.9



	F1	Recall	Precision	Accuracy
MN	61.22	69.27	54.85	56.12
W	58.52	68.97	50.82	51.11
SVD	55.97	59.81	52.6	52.96
TF-IDF	53	54	51	51

Table 2: Plots for comparison of 4 methodologies used in predicting whether a given user will apply to a given job. Dimensions were reduced to 20 for MN Matching, W Regression and SVD.

We showed the comparison between the different methodologies using varying combinations of user descriptions against the job descriptions.

Firstly, one needs to realize that data from Queries is different from data from CV and skills for the following reason: Query is an instantaneous representation of a user's desire to look for a certain kind of job. Also, the keyword searching system of BranchOut restricts the results to containing the keyword that was searched, and this system has an inherent biasness towards a TF-IDF model for prediction (See *Table 2 Queries plot*). On the other hand, information on CV and skills are static and more historical information, which seldom have a direct linkage in keywords in the job being searched, leading to a correspondingly lower TF-IDF scores in the plots with CV and Skills.

To focus our efforts, we discarded plots with regards to Queries, to see if we could discover latent signals that could meaningfully predict a user's interest without having to take into account his explicit expression via his queries.

One can observe from the plots that the MN Method and W regression generally outperformed the baseline TF-IDF and SVD calculations, even in the case of the Queries plot. This is a sign that the latent concepts existing in a user's CV and Skills could act as useful information on his job interests.

Next, between the two methods, the W regression seemed to outperform the MN Method in the case of the Queries and CV, but not in the Skills plot, as the MN Method had higher recall and accuracy in the prediction based on user skills. Our intuition is that the W regression is more of a generic method of similarity calculation, while the MN Method really attempts to look for latent topics. And it thus appears that latent topics are more obvious in the skills set that a user lists about himself.

Finally, we decided to extend the description of a user into his CV and his skill sets, and over there, we saw a slight decrease in performance of the two proposed methods. Perhaps this is indicative of clashes that happen between what a user describes himself on his CV and the skills he list. When a person is given the opportunity to say more about himself, it appears that we seem to know less about him, simply due to him introducing a whole lot of noise.

Nevertheless, interesting insights were obtained. We learnt that the instantaneous query of a user is highly indicative of the job that he is about to apply to, and even TF-IDF performs relatively well over here. However, in terms of finding hidden and latent concepts, the skills a user list about himself offers the best results. In general, the proposed methods of MN Method and W regression outperformed the baseline.

6. Meaningful Inference From Results

We observe some interesting topics that we managed to find from M&N Topic matching. We extracted the words corresponding to the top few weights associated with a particular column of the matrices of M and N. Those that were derived from matrix M refers to the words that a user will use to describe the given topic, while those derived from matrix N refers to the words a job will use to describe the given topic.

The follow topics were example of topics learnt from the CV and Skills description of a user with job description that gave a 0.69 recall.

	User Description	Job Description
Topic on Management and Marketing	Business Corporate Consultant Vendor	Media Decisions Specialists Approach
Topic on Responsibility and Partnership	Responsible Lab Safe Liaison Outreach Travel Financial	Established Proposals Assets Plans Partner Installation
Topic on Performance and Pride	Fine Excellent History Community Marketing	Model First Expectations Productivity NYSE
Topic on Growth and Expertise	Growth Open Intelligence Individuals December	Technology Expertise Engineers University Businesses
Topic on Team	World Team Specialist Identity	Global Energetic Executives Card
Topic on Hierarchy	Principal WWW Requirements Leadership Western Floor	Supervisor Examiners Span Advantage Specifications
Topic on Systems	Strategic Power Systems Building Principal	Supply Delivery Tests Personnel

Topic on Operations	Receptionist Term Limited Wireless Open Contact	Operational Accountant Devices Targets Front Week
Topic on Support	Community Mommy Custom Internal Space	Healthcare Meeting Variety Technology
Topic on Innovation and Control	Film Tour Innovation Facebook Photo	Exceptional Able Required Controls State

Table 3: Meaningful topics extracted from a 20 topic run of MN Method. Note: There were other topics that we were unable to coherently group into topics, but we think nonetheless represent some sort of concept space for matching.

Perhaps more interesting is the capability of our model to use these concepts to predict a wider range of job applications based on transferrable skills. The following table shows some concrete examples where our algorithm has successfully predicted that a user will apply to a given job purely based on skills that are transferrable:

Examples of User Description (Part-Of-Speech extraction)	**Examples of Job Description (Part-Of-Speech extraction)**
The Cashier cashier head cashier phone operator delivery coordinator paint associate dispatcher	*Job Description 1* multi-tasking outlook customer service word ms office
The Bus Operator Bus operator metropolitan transportation mta	*Job Description 2* Truck driver jobs training if ready new career truck industry opportunity highway

Table 4: Examples of users successfully matched to a job description (ie they actually applied on these jobs) using the MN Method.

7. Prediction

Beyond that, we returned to our original problem, and that is to attempt to predict if a given user will click apply on a given job. We first hypothesized that a user's online engagement can act as a baseline if he would apply for a job. We extracted 13 features of a user's online engagement, using engagements that he initiated. These can be found in Appendix.

On top of the baseline, we then incorporated the scores we obtained from the M&N Topic matching as well as the W Regression. We then utilized a non-linear support vector machine to get an improved ability to predict if a user will apply a job. We used libsvm [6] to do the prediction. We used the library's SVM C-SVC with radial basis function: $e^{-\gamma(u-v)^2}$ to do the prediction.

	User Behavior Features	All Features (User as well as User Job Features)
5000 train and 5000 test	Accuracy = 0.607 Recall = 0.503 Precision = 0.588	Accuracy = 0.695 Recall = 0.867 Precision = 0.62

Table 5: Table showing accuracy of prediction using all features via non-linear SVM

From the experiment, one can tell that there is already an ability to tell if a user is going to apply for a job just purely based on his behavior online. However, by incorporating our models for user-job description matching, we were able to get very high recall in predicting if a user will apply to a given job, while maintaining high accuracy.

8. Future Work

With the above, we have identified the following as our next steps. Firstly, we want to improve our topic identification techniques. This could be done via attempting with different objective function and different penalty functions. The hope is that we would be able to extract very clear and distinct topics from BranchOut data. We could also utilize the friendship graph, to see if that could be a better indicator of a user's interest. Also of interest would be experimenting with the temporal axis. We have found through our initial trend mining that time affects the rate of job application, especially at 5pm, 8pm and 1am and on a Friday, and it would be useful if we could account for these spikes in our problem formulation.

With the above results, we would also like to do an online test using BranchOut's system, and measure its impact. This could be measured by looking at the apply rate before and after implementation, which could act as a validation of our recommendation.

9. Conclusion

In this paper, we have aimed to learn how to recommend a job to a user based on transferrable skills of the user. We have learnt to infer these sets of skills using the concept space that exists between a user and job description. Experimental results have also given insights into the types of concepts that can exist, and show that such a modeling provide good potential for good recommendations that will improve user experience on BranchOut. Our models can also be used in various other content matching sites that allow for creative expression on 2 distinct parties, and matching based on common interests must be done, like an advertisement platform.

We also showed that user behavior is an important indicator of whether someone will apply towards a given job. By incorporating user-job description features, we were able to significantly improve (by about 30%) the ability to predict whether a user will apply a job, while maintaining high accuracy.

10. Acknowledgement

We would like to extend our heartfelt gratitude to Jure Leskovec and Andreas Weigend for their support and inspiration throughout the quarter. We also want to thank Rick Marini and his team at BranchOut for providing support and access to their data, without which this paper would not have been possible.

11. Appendix

User engagements initiated by the user himself:

1. Comment Count
2. Create Endorsement Count
3. Friend Request Sent Count
4. Get Intro Count
5. Import Resume Count
6. Job Forward Count
7. Message Reply Count
8. Message Sent Count
9. Search Connection Count
10. Profile View Count
11. Search Job Count
12. Search Resume Count
13. Status Update Count

12. Bibliography

[1] M. Ciaramita, V. Murdock, and V. Plachouras. Online learning from click data for sponsored search. In WWW '08, 2008.

[2] B. Ribeiro-Neto, M. Cristo, P. B. Golgher, and E. Silva de Moura. Impedance coupling in content-targeted advertising. In SIGIR '05, 2005.

[3] C. Wang, D. Zhou, R. Raina, J. Han, D. Fong, G. Badros. Learning Relevance from a Heterogeneous Social Network and Its application in Online Targeting, in SIGIR'11, 2011

[4] I. Konstas, V. Stathopoulos, and J. M. Jose. On Social Networks And Collaborative Recommendation. In SIGIR '09, 2009.

[5] Z. Wen and C.-Y. Lin. On the quality of inferring interests from social neighbors. In KDD '10, 2010

[6] C. Chang, C. Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm