Modeling Professional Similarity by mining Professional Career Trajectories*

Ye Xu[1], Zang Li[2], Abhishek Gupta[2], Ahmet Bugdayci[2], Anmol Bhasin[2]
[1]Dartmouth College, ye@cs.dartmouth.edu
[2]LinkedIn Corporation, {znli, agupta, abugdayci,abhasin}@linkedin.com

ABSTRACT

For decades large corporations as well as labor placement services have maintained extensive yet static resume databanks. Online professional networks like LinkedIn have taken these resume databanks to a dynamic, constantly updated and massive scale professional profile dataset spanning career records from hundreds of industries, millions of companies and hundreds of millions of people worldwide. Using this professional profile dataset, this paper attempts to model profiles of individuals as a sequence of positions held by them as a time-series of nodes, each of which represents one particular position or job experience in the individual's career trajectory. These career trajectory models can be employed in various utility applications including career trajectory planning for students in schools & universities using knowledge inferred from real world career outcomes. They can also be employed for decoding sequences to uncover paths leading to certain professional milestones from a user's current professional status.

We deploy the proposed technique to ascertain *professional similarity* between two individuals by developing a similarity measure *SimCareers* (Similar Career Paths). The measure employs sequence alignment between two career trajectories to quantify professional similarity between career paths. To the best of our knowledge, *SimCareers* is the first framework to model professional similarity between two people taking account their career trajectory information. We posit, that using the temporal and structural features of a career trajectory for modeling profile similarity is a far more superior approach than using similarity measures on semi-structured attribute representation of a profile for this application. We validate our hypothesis by extensive quantitative evaluations on a gold dataset of similar profiles generated from recruiting activity logs from actual recruiters using LinkedIn. In addition, we show significant improvements in engagement by running an A/B test on a real-world application called *Similar Profiles* on LinkedIn, world's largest online professional network.

*This work was done when Ye Xu interned in LinkedIn.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
KDD'14, August 24–27, 2014, New York, NY, USA.
Copyright 2014 ACM 978-1-4503-2956-9/14/08 ...$15.00..

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Data Mining

Keywords

Social Networks, Similarity, User Profiles, Career Paths

1. INTRODUCTION

Online professional networks are coming of age and are becoming an important tool for maintaining one's professional profile of record and discovering new career opportunities. LinkedIn maintains the professional profile for over 238 million members. With such a large number of profiles, quality profile discovery at scale becomes a challenging problem.

Recruiting is a massive use case exercised by premium users of LinkedIn. *Similar Profiles Recommender System* helps recruiters and hiring managers discover other similar quality talent by pivoting of a model user profile. It models each member profile, as illustrated in Fig.2, by extracting a labeled bags of canonicalized keywords from profile fields such as summary, skills, companies worked at, schools attended, job titles etc. Given the bags of keywords representation for user profiles, *Similar Profiles* measures the similarity between pairs of LinkedIn members by matching keywords across field pairings, and then comes up with a normalized similarity score. While relatively powerful, this approach does not leverage a critical aspect, which is fundamental to assessing professional similarity - the temporal information encoded in series of positions held by the individuals through their careers.

Let us take Fig.1 as a motivating example[1] to highlight the importance of using Career Trajectory information for finding similar people. Here, we present the top several members who are most similar to a particular user as returned by *Similar Profiles*. When we peruse through the returned list, we find that both Member 1 (ranked 1^{st}) and Member 7 (ranked 7^{th}) are similar to the model profile in terms of overall semantic keyword similarity. However, if we examine the appeared time of matched keywords in both profiles, we observe that the matched keywords (e.g., "Data analysis") between model profile and Member 7 (ranked 7^{th}) both appear in their most recent position, while the matched keywords (e.g., "Data Mining") between model profile and Member 1 (ranked 1^{st}) appear in past positions. From a recruiting standpoint, it is opined that recent experience is more applicable to profile similarity in addition to length of the experience with particular skillsets. Hence, it is intuitive to seek that the model profile is more similar to Member 7 (ranked 7^{th}) than Member 1 (ranked 1^{st}) in terms of career trajectory. Since the current *Similar Profiles Recommender System*

[1]For the purpose of privacy protection, we anonymize the name and profile pictures of returned results.

Figure 1: An example of *Similar Profiles*. All the returned results are similar in terms of keyword matching. However, because *Similar Profiles* fails to consider career trajectory information when calculating the profile similarity, a portion of returned results are not similar in terms of career trajectory.

ignores temporal information, we miss this subtle yet important signal.

To address this problem, we propose the *SimCareers* (Similar Career Paths) framework, a new approach to modeling LinkedIn member profiles by leveraging the concept of career trajectory. The *SimCareers* treats every individual member profile as a sequence of nodes, each of which records all information of the position, such as company, title, industry, time duration, and keyword summary. Then, based on the profile modeling method, the similarity between two member profiles is calculated by aligning the two sequences of nodes. At the node level, similarity is ascertained by using a generalized linear model but other approaches could be easily substituted.

The aforementioned approach is an effective way to model professional profiles, under which a comprehensive view of member's professional information is exhibited in a time-series manner. Its most immediate application is to help improve *Similar Profiles Recommender System*. This framework can be extended to help professionals do career planning. For example, by comparing career paths between young professionals and the early stage profiles from those of more successful senior individuals, we can give people a look-ahead of possible future career trajectories based on where they currently are. This can help them decide which school to choose, area of specialization to pursue and skills to acquire in order to achieve the desired outcome.

In summary, the key contributions of this paper are as follow: (1) To the best of our knowledge, *SimCareers* is the first framework that models professional profiles as time-series of career positions for any online professional network or resume databank. (2) We propose a similarity measure as a sequence alignment exercised over professional user profiles to ascertain professional similarity between two professionals from a recruiting standpoint. (3) We reveal and validate the important insight that career trajectory information is of paramount importance in modeling similarity over professional profiles. The validation is facilitated by running real-world *Similar Profiles* application experiments on the world's largest professional network - LinkedIn.

2. BACKGROUND

Similar Profiles Recommender System is used in LinkedIn for profile modeling and quality candidate discovery. In addition to serving as an independent product, it also powers *People You May Want to Hire* which is a personalized candidate discovery engine for recruiters that takes into account all of the context with regards to recruiting activity [21].

Formally speaking, the *Similar Profiles Recommender Problem* can be defined as follows:

PROBLEM 1 (SIMILAR PROFILES RECOMMENDER PROBLEM). *Given all LinkedIn members $u \in \mathcal{U}$, where each member u is associated with a member profile f^u, and a source member s, the Similar Profiles Recommender System outputs a list of k target members $t_1, t_2, ..., t_k$ who have top-k highest similarity scores to the source member, and the list of members is ranked by the similarity value in a descending order.*

Similar Profiles algorithm works as follows: (An illustration is given in Fig. 2.) It models member profiles by extracting keywords from each field in member's LinkedIn profile. A few example fields are summary, skills, current position summary, past position summary, companies worked at, schools attended, etc. These fields

are placed on the member's profile on LinkedIn where members have manually entered free form text. The extracted keywords form lists of bags of keywords. We use cosine-similarity to find similarity scores for field-pairs corresponding to the two member profiles in consideration. An overall similarity score is calculated using a weighted linear combination of these scores to get the final *Similar Profiles* similarity measure between the two members. This measure is symmetric and is normalized using a logit function, so that it can be interpreted as a probability of being similar. Here, weights reflect the relative importance of field pairs that are matched. These weights are learned by fitting a logistic regression model on training data obtained from active recruiter usage of *Similar Profiles* product.

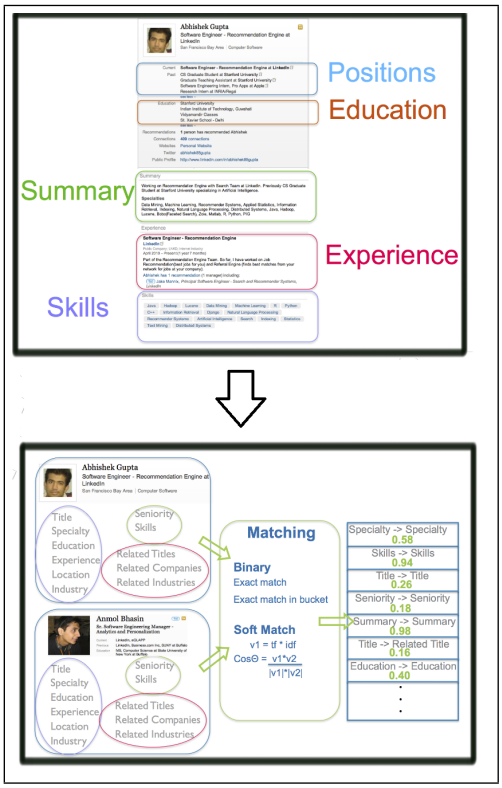

Figure 2: The procedure of *Similar Profiles*. First, keywords are extracted for every field from LinkedIn profile page for each member. Then, given two bags of keywords, we match keywords at the field level. Finally, we come up with a similarity score.

3. SIMILAR CAREER PATHS

In what follows, we give the description of Similar Career Paths framework, which models LinkedIn member profiles as career sequences. We use sequence alignment to measure overall profile similarity. For every node in the sequence, we use keyword based matching to evaluate node-level similarity.

3.1 Problem Description

For each LinkedIn member, we maintain a profile webpage[2] that records his/her professional information, e.g., title, company, seniority, overall summary, experience, skills and expertise, education, courses, languages, additional personal information, connections, groups etc. An example is shown in Fig.3. Given the profile

[2]http://www.linkedin.com/profile/

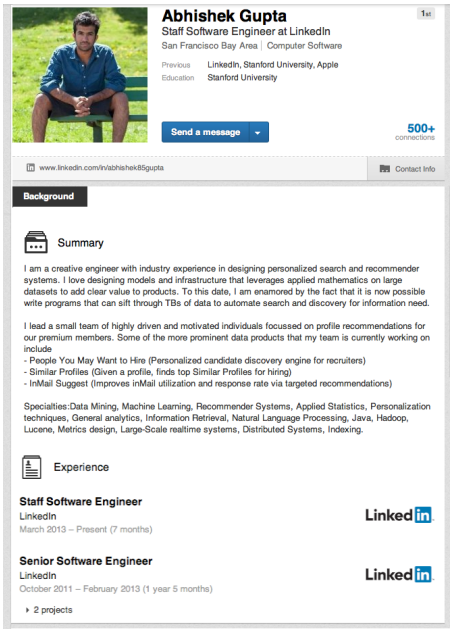

Figure 3: An example of LinkedIn member profile page: it contains professional information of the member such as job tittle, seniority, company, location, education, summary, experience etc. All these information will be extracted to learn the similarity metric between two members.

pages from two LinkedIn members, we aim to measure the similarity between them. In other words, all the information with regards to member profile can be used to calculate the similarity score between them.

3.2 Model Member Profiles

In what follows, we describe how we model LinkedIn member profiles. Like many other online social networks [7, 11, 4], *Similar Profiles*, models member profiles purely by keywords matching. It ignores the temporal order of keywords and thus fails to capture the career trajectory information, which is extremely important in a professional network.

To solve this problem, we introduce the concept of timeline. In *SimCareers*, each member profile is treated as a time sequence of nodes. In this modeling scheme, firstly, we provide a clear view for each member's career trajectory. Secondly, we explain how we divide the whole bag of keywords in the original *Similar Profiles* into finer granularity. Finally, we also describe how the temporal factor is taken into account in the similarity computation. Intuitively, the matched keywords at the closest timestamp implies that the profiles are more similar with respect to more recent work experience. Hence, it should lead to higher overall similarity. More concretely, we employ the following schemes to model the node corresponding to a job position.

Sequence of Positions: In this scheme, each node represents one particular position of the member's professional experience, e.g., position summary, company, title, seniority, industry, job function, and time durations of the position. An example is given in Fig.4.

Sequence of Compositions: In this scheme, we still model each LinkedIn member's profile using a sequence of node. For each node, in addition to using position information mentioned before, we also incorporate transition information associated with the given

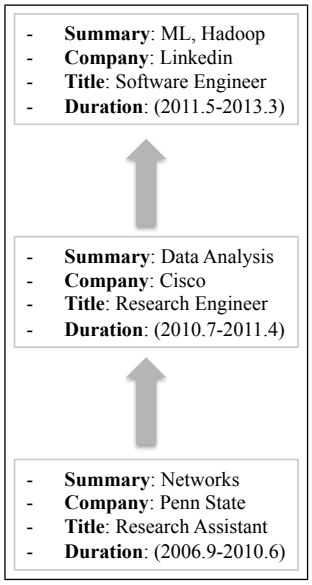

Figure 4: Sequence of positions modeling method: each node in the sequence contains information about the position, such as position summary, company, title, seniority, and time durations.

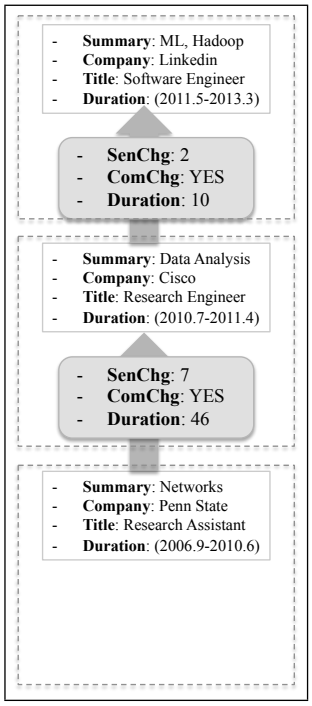

Figure 5: Sequence of compositions modeling method: each node in the sequence is the composition of position information and transition related information. Incorporating transition information enhances the representation of each node. Note that the first composition node is associated with the earliest position. It has no previous positions, thus this node does not have any transition related features.

position and the previous one. In other words, the position information, along with transition related information, together composes a node. Transition information, (i.e., whether title changes in this transition, whether company changes, how the seniority changes, and the time of years used in this transition) enhances the representation of the *Sequence of Compositions* modeling scheme by further disclosing information of the changing trend between previous and the given position. An example is given in Fig. 5.

3.3 Model Career Path Similarity

In this work, we design a similarity metric based on the sequence of nodes modeling scheme. Generally speaking, given two sequences of nodes (profiles), we conduct sequence alignment to calculate the similarity between them. Sequence alignment [22] originated in bioinformatics domain. It can identify the similarity between two sequences of DNA, RNA or protein by finding an optimal way to arrange the node-level alignment from two sequences respectively. In our profile similarity learning scenario, we need to match and align career nodes from two career sequences (i.e., member profiles) according to the similarity between them. Thus, this is naturally associated with the classic sequence alignment problem. When it comes to the specific alignment between each pair of node, we propose a node-level similarity metric. We discuss the node-level Similarity metric in more detail in Section 3.3.2.

3.3.1 Profile-Level Similarity by Sequence Alignment

We aim to evaluate the similarity between two career paths – two sequences of nodes. Each node here is a representation of one particular work experience. For overall similarity score, we want all the work experience in the career sequence to contribute. In order to compute overall similarity between two career sequences, we decompose the score into the sum of the similarity between several pairs of aligned nodes from each of the two sequences respectively [8]. We now need a scheme to find an optimal alignment between pairs of nodes from the two career paths. This naturally relates to the classic sequence alignment problem.

In sequence alignment algorithm [16], the sequence level similarity is measured by calculating the sum of the optimal alignment of node pairs. We conduct the sequence alignment in a local manner [8]. The two sequences are aligned incrementally. The sequence alignment scheme can be formulated as a dynamic programming procedure. Suppose, we have two career sequences $\mathbf{P_1} = [\mathbf{X_1}, \mathbf{X_2}, , \mathbf{X_m}]$ and $\mathbf{P_2} = [\mathbf{Y_1}, \mathbf{Y_2}, , \mathbf{Y_n}]$. ($\mathbf{X_i}$ and $\mathbf{Y_j}$ are position/composition nodes from two career sequences respectively.) Let us further imagine that we have come to the step of aligning subsequences $P_1[1 : i-1]$ and subsequence $P_2[1 : j-1]$. (In other words, shorter subsequences have been aligned previously.) The subsequences $P_1[1 : i]$ and $P_2[1 : j]$ can be aligned in three ways according to the following cases:

- The node $\mathbf{X_i}$ is similar to node $\mathbf{Y_j}$. This leads to this pair of positions being aligned and results in an overall increase in sequence similarity score as contributed by this node similarity value. Here, $\mathbf{P_1}[1 : i]$ represents the subsequence $\mathbf{X_1}$, $\mathbf{X_2}$, .., $\mathbf{X_i}$ from career sequence $\mathbf{P_1}$.

- The node $\mathbf{X_i}$ is not much similar to node $\mathbf{Y_j}$. Thus, $\mathbf{X_i}$ will be skipped. Note that although a node is allowed to be skipped during sequence alignment, we encourage contiguous alignment for the purpose of career path completeness. Therefore, we impose a gap penalty on sequence level similarity score when skipping a node. (We will further discuss this point in later section)

- And vice versa, if the node $\mathbf{X_i}$ is not much similar to node $\mathbf{Y_j}$, we impose the same gap penalty.

It is worth noting that the position-level similarity function that we employ is symmetric. Hence, $S^{node}(X\_i, Y\_j)$ is the same as $S^{node}(Y\_j, X\_i)$. More formally, given the above two career sequences $\mathbf{P_1}$ and $\mathbf{P_2}$, the similarity between two career sequences can be solved using the following scheme:

$$S^{seq}(\mathbf{P_1}[1:i], \mathbf{P_2}[1:j]) = \max \begin{cases} S^{seq}(\mathbf{P_1}[1:i-1], \mathbf{P_2}[1:j-1]) + S^{node}(\mathbf{X_i}, \mathbf{Y_j}) \\ S^{seq}(\mathbf{P_1}[1:i-1], \mathbf{P_2}[1:j]) - \lambda \\ S^{seq}(\mathbf{P_1}[1:i], \mathbf{P_2}[1:j-1]) - \lambda \end{cases} \quad (1)$$

Therein, S^{seq} is the similarity function at the career sequence level, S^{node} is the similarity function at the position/composition node level, and λ is the gap penalty parameter. We will discuss S^{node} and λ in later sections.

3.3.2 Node-Level Similarity Model

The *SimCareers* framework aligns two career sequences by incrementally aligning two position/composition nodes or skipping the node in either of the two sequences. In order to compute S^{seq}, we need to first define S^{node}. We propose to learn a similarity model at the node level by using Logistic Regression model [3]. In what follows, we will discuss the node-level similarity model in detail.

Data and Labels. We first extract position/composition node data features from LinkedIn member profile. We gather training data from active recruiter usage of the *Similar Profiles Recommender System*. Recruiters use this recommender system to discover more profiles similar to the query[3]. *Similar Profiles Recommender System* that is powered by *SimProfiles* algorithm, as discussed in Section 2, can guarantee the returned candidate profiles are similar to the query profile. Once recruiters find desired candidates, the most common way for them to reach out to candidates is via *InMail*. InMail is a paid product provided by LinkedIn that allows recruiters to reach out to members outside their existing network. In the absence of ground truth data for position similarity, we employ the following heuristic to gather quality training data for position similarity model.

We only consider profile-pairs that were discovered via *Similar Profiles Recommender System* and then contacted by recruiters. In other words, we only look at *inMails* sent as a result of *Similar Profiles Recommender* search. We regard the current position of such profile pairs, the original profile looked at and the subsequent one contacted, as similar. The assumption used here in collecting data is that if a recruiter contacts this newly discovered profile immediately after discovering the original one then presumably at-least the current position of these profiles is similar. Empirically speaking, most recruiters in LinkedIn behaves in this manner. With this scheme, we sample about $300,000$ candidate profile pairs with a non-empty current position from data collected over all LinkedIn Recruiter accounts within 3 months (from March 2013 to May 2013). Note that the intuition of introducing recruiter contact here is that current Similar Profiles Recommender System is not perfectly accurate and could return some dissimilar pairs. Thus by examining recruiter contact, we can refine the quality and purity of data.Such scheme, along with the amount of data processed, can guarantee the quality of the collected data.

As a further refinement for model training, we only consider those profile-pairs as positives where more than 3 recruiters contact the profiles discovered via *Similar Profiles Recommender System*. The intuition is that if enough recruiters contact the newly discovered profile then this profile pair is more likely to be similar. Amongst the rest of the recommended profiles, all the ones that were ranked higher but were not reached out to be any recruiter are regarded as negative profile-pairs for the purpose of model training for node-level similarity. After employing this additional constraint, we get about $80,000$ profile-pairs as positive labels for position similarity model training and $220,000$ negative labels. During the model training, we use all $80,000$ positive position pairs and randomly sample $80,000$ negative position pairs from the $220,000$ candidate negative labels.

Features. Based on the discussion in Section 3.2, we have two schemes to model member profiles in *Similar Career*: (1) sequence of positions; and (2) sequence of compositions. For each of the two methods, we consider different feature sets to train the corresponding node level similarity model.

For the sequence of positions modeling scheme, we extract the following features from each pair of positions:

- *CurrentTitle:* The feature indicates whether the two positions share the same title name.
- *CurrentCompany:* The feature indicates whether the two positions are from the same company.
- *CurrentCompanySize:* At LinkedIn, each company is categorized into a bucket based its employee size (e.g., less than 100 employees, 100 to 1000 employees, 1000 to 10000 employees, and more than 10000 employees.) This feature measures the difference between two positions in the aspect of company size.
- *CurrentIndustry:* This feature indicates whether the two positions are from the same industry. E.g., of some industries are Internet, Venture Capital, Hardware.
- *CurrentFunctions:* This feature indicates whether the two positions have the same job function. E.g., of some job functions are Engineering, HR.
- *JobSeniority:* This is a derived feature that takes into account overall work experience in terms of titles held in the past, companies worked at and number of years of work experience. E.g., Director of Engineer at LinkedIn would have a higher seniority than Engineering Manager at LinkedIn.
- *CurrentPositionSummary:* This feature extracts key words from the position summary field.
- *TittleSim:* This feature indicates whether the titles of the two positions are similar or not. E.g., Applied Research Engineer and Data Scientist are similar titles.
- *CompanySim:* This feature indicates whether the companies of the two positions are similar or not. E.g., from an employability standpoint, LinkedIn, Facebook and Google are considered similar.
- *IndustrySim:* This feature indicates whether the industries of the two positions are similar or not. E.g., Computer Software and Internet are more similar to each other than Internet and Civil Engineering.
- *Duration:* This feature describes the difference of time durations between the two positions.

[3]Here, the query member profile is often the ideal candidate they have found or already hired.

It deserves mentioning that we normalize each feature value and make them between 0 and 1.

For the sequence of composition nodes modeling scheme, we treat the node as a composition of position and its associated transition. Therefore, in addition to the above mentioned position related features, we also consider a few transition related features, namely, features from the transition associated with the position and its immediately previous position. The following additional features are extracted for each pair of composition nodes:

- *IsSameCompany:* This feature indicates whether both of the two transitions happen within the same company or not.
- *IsSameIndustry:* This feature indicates whether both of the two transitions happen within the same industry or not.
- *SeniorityChange:* This feature indicates whether the seniorities changed for both of the two transitions.
- *TitleChange:* This feature indicates whether the titles change for both of the two transitions.
- *TimeGap:* This feature indicates the difference of how many year durations pass during the two transitions.

Similar to position node features, we also normalize the feature values.

Model: A variety of classification methods can be used to train the similarity model given the data and labels. Note that, we aim to propose a general framework to train the node level similarity model, and our setup is not limited in any way to any particular training algorithm. Without loss of generality, in this work, we employ Logistic Regression (LR) algorithm [3] for model training. In logistic regression model, the logit of the probability of relevance of the outcome is modeled as a linear function of feature values. The model is trained via maximum likelihood estimation. Due to its effectiveness, logistic regression is widely used in many application domains such as recommender systems [2, 15] and advertisement prediction [19, 5]. Logistic regression predicts the outcome of a value between 0 and 1 according to the feature values. In our scenario, the similarity value is bounded between 0 and 1. Therefore, logistic regression is ideally suitable to train the node level similarity model. Furthermore, since existing keyword based *Similar Profiles* also employs logistic regression so this mechanism provides an easy way to combine these 2 scores. We discuss a need to do so in section 3.5.

3.4 Other Factors in the Framework

In this section, we discuss the parameters used in the *SimCareer* framework.

3.4.1 Gap Penalty

In the process of aligning career sequences, the dissimilar position/composition nodes are allowed to be skipped. However, for the purpose of career path completeness, contiguous alignments are more desirable. Therefore, to encourage contiguity in career sequence alignment, we introduce the idea of gap penalty, as shown in Eq.1. The sequence alignment is computed in an incremental manner. As we encounter node pair $\mathbf{X_i}$ and $\mathbf{Y_j}$, if the node level similarity between $\mathbf{X_i}$ and $\mathbf{Y_j}$ is high enough as determined by the *Dynamic Programming* algorithm shown in Eq.1, we continue to align them. However, if $\mathbf{X_i}$ is dissimilar to node $\mathbf{Y_j}$ in sequence $\mathbf{P_2}$, we allow the possibility of skipping the alignment of the node with a penalty called *Gap Penalty* because it impairs the contiguity in sequence alignment.

In this setting, value of the gap penalty parameter λ needs to be determined. If the value of λ is too small, many nodes can be skipped during alignment and the contiguity will be impaired. On the other hand, if it is too large, no skip could happen during alignment. As disclosed in [22], in absence of any prior knowledge, grid search is a reasonable way to determine the *Gap Penalty* parameter λ. We apply $10-$fold cross validation to determine the optimal gap penalty value. It deserves mentioning that the node level similarity score in our framework is bounded between 0 and 1. Therefore, it is relatively convenient to guess candidate values for λ since it should have the same order of magnitude. In section 4.1.3, we will show that the performance of the proposed *SimCareers* is not sensitive to the λ value.

3.4.2 Position Recency and Duration

SimCareers aims to detect the similarity between two career paths under the scenario of professional social networks. In discovering professional candidates, more recent experiences are more highly valued than old experiences. Furthermore, longer period of job experiences are more convincing than shorter ones. Therefore, the duration and the recency of positions are worth considering when we calculate the similarity between career sequences.

As discussed in section 3.3.2, we try to incorporate duration information as a feature when training the node level similarity model. However, note that the training data used to calculate position similarity is for latest positions only. This implies that all these job experiences are ongoing, and we cannot reliably predict when these current job positions will terminate for each member. Therefore, with our scheme of gathering training data it is inaccurate to apply the duration of current positions as a feature to train the node level similarity model. By the same argument, the recency for current positions are always 0 (i.e., 0 years from today) in our training data. Due to lack of an automated way to generate large amounts of training data, we propose to incorporate these signals as boosts to the node-level similarity for both sequence of positions and sequence of compositions method.

As mentioned above, we incentivize alignment on more recent positions and longer duration nodes. More specifically, while computing similarity between two career sequences, we impose different weights on aligned position/composition nodes according to the recency and duration of the aligned node pair. Thus, the modified sequence alignment can be formulated as follows:

$$S^{seq}(\mathbf{P_1}[1:i], \mathbf{P_2}[1:j]) =$$
$$\max \begin{cases} S^{seq}(\mathbf{P_1}[1:i-1], \mathbf{P_2}[1:j-1]) + w(i,j) S^{node}(\mathbf{X_i}, \mathbf{Y_j}) \\ S^{seq}(\mathbf{P_1}[1:i-1], \mathbf{P_2}[1:j]) - \lambda \\ S^{seq}(\mathbf{P_1}[1:i], \mathbf{P_2}[1:j-1]) - \lambda \end{cases}$$

Therein, $w(i,j)$ is the weight for aligned node pair $\mathbf{X_i}$ and $\mathbf{Y_j}$. Intuitively, longer duration and more recent nodes should have larger weight so that they can impose stronger influence on the final sequence level similarity score. Additionally, if the difference between two nodes in terms of recency and duration is smaller, we assume that the two nodes are more similar. Thus, in our scheme this weight incorporates four terms: (i) The duration difference between two nodes; (ii)The recency difference between two nodes; (iii) The duration sum of two nodes; (iv) The recency sum of two nodes.

In this work, we apply the Half-life in exponential decay [13] as the weight. Half-life exponential decay is widely used to describe a quantity undergoing exponential decay with the following formulation:

$$N(t) = e^{-\tau \cdot t} \tag{2}$$

Therein, τ is the mean lifetime parameter. Using Eq.2, the weight $w(i,j)$ in sequence alignment formulation can be written as follow:

$$w(i,j) = e^{-\tau_1|r_i - r_j|} e^{-\tau_2|d_i - d_j|} e^{-\tau_3(r_i + r_j)} (1 - e^{-\tau_4(d_i + d_j)}) \quad (3)$$

Therein, d_i is the duration of node $\mathbf{X_i}$ and r_i is the recency of node $\mathbf{X_i}$. Note that we encourage small difference of duration and recency, small value of recency but large value of duration. The mean lifetime parameters τ_1, τ_2, τ_3 and τ_4 are determined by using $10-$fold cross validation.

By assigning higher weights to recent or longer duration positions, their importance in discovering the similarity between professional sequences are highlighted.

3.4.3 Career Sequence Length Bias

Various LinkedIn members have different lengths for their career sequences. A very small fraction of LinkedIn members have a large sequence length due to their long career. Note that when recruiters are discovering profiles for hiring, member's more recent positions are given more importance. With this in mind, we limit the number of past positions that we consider for overall profile similarity in the *SimCareers* framework. In other words, if a member has more than L position/composition nodes in his/her career sequence, we only consider the latest L nodes for analysis.

Choosing a proper L value is important. If it is too small, important position nodes are missed. On the other hand, a large L value makes *SimCareers* consider unimportant position nodes, and also may lead to inaccurate results. Instead of using cross validation, we analyze the sequence length distribution over all LinkedIn members and pick the value for L. A significant proportion of our members have at-most 9 positions, so we choose $L = 9$.

3.5 Incorporate Similar Profiles

In our final scoring, we incorporate the existing *Similar Profiles* score into the proposed *SimCareers* framework. In what follows, we explain the motivation of considering this keyword based score.

3.5.1 Sparse Position Problem

On our member's LinkedIn profiles, it is common for members to have comprehensive and detailed descriptions in their overall summary field. However, when it comes to the position-level summary section there is huge variability in the amount of content our members provide, from hundreds of keywords to nothing at all. Furthermore, overall summary provides important information about member's skills and speciality as it stands today. In *Similar Profiles Recommender System*, this is one of the most important features. Hence, incorporating keyword based similarity score is desirable.

3.5.2 Ensemble Approach

After generating member similarity score using two separate signals, i.e., the approach discussed above and the *Similar Profiles* score, the next step is to create a unified scheme that incorporates the two signals.

Ensemble learning [25] that uses multiple models to obtain better predictive performance naturally fits our scenario. Because of its powerful generalization ability, ensemble learning has been used in many applications [17]. Based on how to combine the results from each weak classifier, ensemble methods can be divided into a few categories: bagging, boosting, stacking, and cascading [25]. In this paper, we apply Bagging scheme to incorporate *Similar Profiles* score into the proposed *SimCareers* due to its simplity and efficiency. Specifically speaking, we apply the weighted combination of *SimilarProfiles* score and sequence alignment score using the method discussed in section 3.3. Note that *Similar Profiles* score is normalized between 0 and 1. However, the sequence alignment score is not because most people have career sequences of with more than 1 position. Thus, we normalize the sequence alignment score by dividing by the total number of sequence length of source person. Fig.6 shows the probability density function (PDF) of normalized sequence alignment score and *Similar Profiles* score over *LinkedIn member dataset* from March 2013 to May 2013 (i.e., the current position data we use for training node level similarity model). It implies that after normalization, sequence alignment score have similar distribution to *Similar Profiles* score, and thus validates that applying this ensemble approach in *SimCareers* is reasonable.

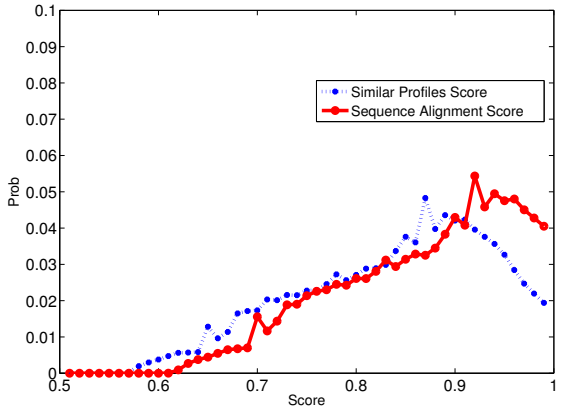

Figure 6: The Probability Density Function (PDF) of sequence alignment score and *Similar Profiles* score over *LinkedIn member dataset* within three months. Note that LinkedIn Member data is from *Similar Profiles Recommender System*.

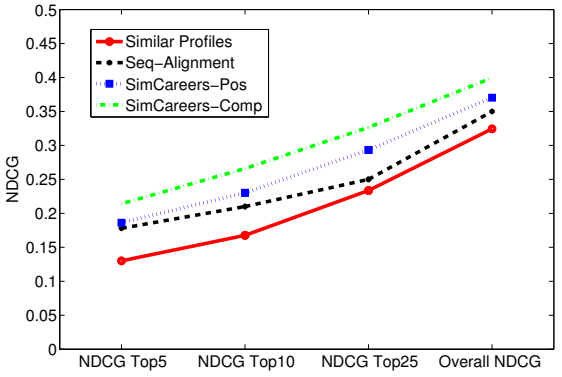

Figure 7: The NDCG scores of each method: the two *SimCareers* framework achieves much better NDCG scores than *Similar Profiles*.

4. EVALUATION

In what follows, we discuss the evaluations for the proposed *SimCareers* framework. In the first evaluation approach, we apply an offline measure of Normalized Discounted Cumulative Gain (NDCG) [10]. This measure is widely used to gauge the effectiveness of web search engines or recommender systems. Then, we

also test the effectiveness of *SimCareers* by performing online AB tests and measuring the impact on our business metrics.

4.1 Offline Evaluation

The offline evaluation scheme offers a reasonable and easily repeatable mechanism for verifying performance and tuning model parameters. As a classic metric for search engine and recommender system, NDCG is used to measure the accuracy of the proposed *SimCareer* framework.

4.1.1 Experimental Setup

Dataset: We get the testing data from *Similar Profiles Recommender System* on LinkedIn. As discussed in section 3, once a recruiter account uses this recommender system to discover candidates based on a source profile, the source profile and the returned 100 recommended candidates are recorded. Note that here we only look at the candidates returned from *Similar Profiles Recommender System*. The results are shown to recruiters in descending order of *Similar Profiles* score. To demonstrate the effectiveness of our proposed framework, we use *SimCareers* to evaluate the similarity between each of the 100 pairs (i.e., source profile and each of the 100 returned results) and obtain a reranked list. Note that when calculating NDCG metric, the ground truth order of the 100 profiles is needed. Here, we employ *InMail* sending and receiving to infer this ground truth.

Similar to the scheme discussed in section 3, if after seeing Similar Profiles results for a source profile the recruiter contacts one or more of these recommended results, then we regard the source profile and the profiles that were contacted as *Similar*. As a further refinement, we only consider those cases as positives where at least 3 recruiters contacted the recommended profile. Meanwhile, within the top 100 results, the recommended profiles which were ranked higher, were seen by the recruiter and were not contacted are regarded as *Not Similar* for the purpose of collecting labeled dataset. All other recommended profiles are ignored from the labeled dataset.

In this experiment, we use data collected over all LinkedIn Recruiter accounts within one month (June 2013) for testing. We denote the base *Similar Profiles* score by *sp* and sequence alignment scores derived by *SimCareers* by *sc*. As discussed in section 3.5.2, we do a linear combination of *sp* and *sc* score via parameter δ i.e. unified relevance score = *sp* + δ * *sc*. We report offline numbers for $\delta = 1.0$.

Metric: We use Normalized Discounted Cumulative Gain (NDCG) [10], a classic metric used for evaluating search and recommender systems. The NDCG score is calculated using the following formulation:

$$nDCG = \frac{DCG}{IDCG} \tag{4}$$

Here, Discounted Cumulative Gain (DCG) is defined as follows,

$$DCG = rel(1) + \sum_{i=2}^{N} \frac{rel(i)}{log_2 i} \tag{5}$$

Therein, $rel(i)$ is the relevance score of position i. In our scenario, if the i^{th} returned profile is similar in terms of inMail validation, $rel(i) = 1$. Otherwise, $rel(i) = 0$. $IDCG$ in Eq.4 is calculated using the same equation as DCG based on the ground truth ranking list.

Similar Profiles Recommender System is used by recruiters for discovering quality candidates given a source profile. Intuitively, doing well on the top results is better than doing well on the bottom results. Thus, the quality of top returned results is more important. To demonstrate the effectiveness of *SimCareers* we compute

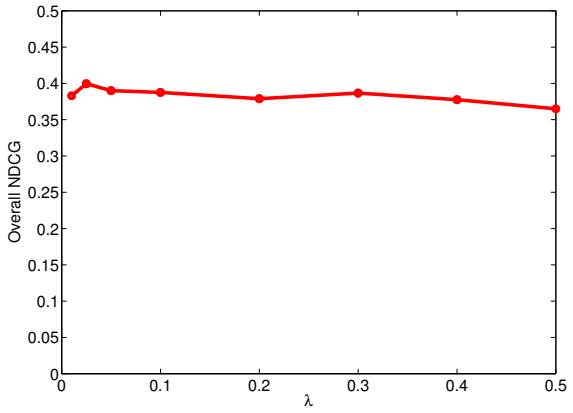

Figure 8: The NDCG scores of *SimCareers-Comp* under different λ values: (Other parameters are fixed here.) The results demonstrate that the proposed *SimCareers* framework is not sensitive to gap penalty parameter λ.

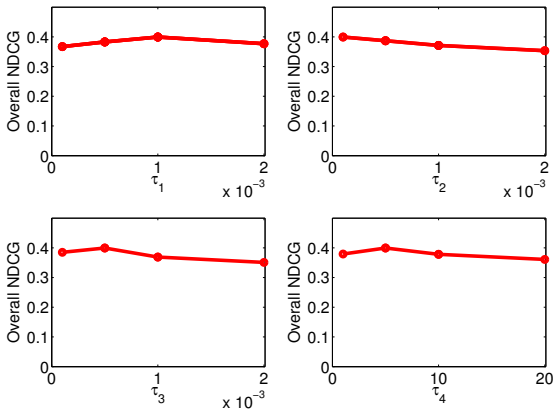

Figure 9: The NDCG scores of *SimCareers-Comp* under different τ values: each time we vary one parameter while fixing the rest.

NDCG top5, NDCG top10, and NDCG top25. (i.e., let $N = 5$, 10, and 25 respectively in Eq.5.) in addition to overall NDCG (i.e., let $N = 100$ in Eq.5).

Baseline: We use the ranking results from *Similar Profiles* as the baseline to evaluate *SimCareers*. Moreover, we compute the NDCG score using purely the sequence alignment method (*Seq-Alignment*), i.e., without resorting to the ensemble approach mentioned in Section 3.5.2. To validate the importance of using transition related information when modeling career sequence similarity, we calculate the NDCG score for sequence of positions modeling method (*SimCareers-Pos*) and sequence of compositions modeling method (*SimCareers-Comp*) respectively.

4.1.2 Experimental Results

Fig.7 shows the results of NDCG top5, NDCG top10, NDCG top25, and overall NDCG score for *SimCareers-Pos*, *SimCareers-Comp*, *Seq-Alignment* and *Similar Profiles*. These numbers validate our intuition that capturing temporal information of career tra-

jectory is crucial in computing profile similarity for hiring. More specifically, pairwise t-tests at 95% significance level indicate that the *SimCareers* framework is significantly better than *Similar Profiles* under all the four NDCG scores. Meanwhile, the two *SimCareers* framework achieves high NDCG scores than *Seq-Alignment*, demonstrating the effectiveness of the ensemble approach. By comparing the curves of *Seq-Alignment* and two *SimCareers* methods, we can reach the conclusion that incorporating keyword based similarity score is effective and desirable. It deserves mentioning that compared with overall NDCG, the difference between *SimCareers* and *Similar Profiles* under NDCG top5 and NDCG top10 is larger. It means in the more realistic setting where top recommended profiles will end up getting contacted more often, the proposed *SimCareers* will perform even better. Note that under all the four metrics, *SimCareers-Comp* obtains higher NDCG scores than *SimCareers-Pos*. Incorporating transition related information into the position node provides a more comprehensive description and captures the dynamic changes of the work experience nodes, thus leading to higher quality *Similar Profiles*.

4.1.3 Parameter Insensitivity

To further demonstrate that *SimCareers* is not sensitive to the parameters we define in the model, i.e., the gap penalty λ and mean lifetime parameter τ_1, τ_2, τ_3, and τ_4, we report the overall NDCG score of *SimCareers-Comp* under different parameter settings. Each time we vary one parameter, and fix the rest. The NDCG scores under different λ values are shown in Fig.8, and the results under different mean lifetime parameter values are shown in Fig.9. The two figures indicate that the *SimCareers* framework is not sensitive to these parameters. It is worth noting that the dataset used for parameter estimation is a mutually disjoint dataset from the training dataset. This dataset spans March 2013 to May 2013 and serves as a validation dataset to tune these additional parameters. The difference in dataset accounts for the difference in absolute value of overall NDCG in Fig. 7.

4.2 A/B Test

We performed online AB tests[12] for a few variants of *SimCareers* model on independent segments of LinkedIn member populations. When combined with offline NDCG score for the similarity model, A/B Test provides for a deeper understanding of user behavior. We focus on the following metric for the online evaluation:

- Profile Views: This describes the profile views generated via *Similar Profiles Recommender System* from recruiters looking to hire talent. For every candidate that a recruiter discovers, we allow him to discover more profiles via *Similar Profiles* recommendations. Viewing one or more of these recommended results indicates that the recommendations are relevant.

- InMails : This describes contact between recruiters and the candidate discovered via *Similar Profiles Recommender System*. The assumption is that the more candidates recruiters reach out to via recommendations the more similar the recommendations are to the source profile that lead them to discover newer profiles.

Since combining *Similar Profiles* score and sequence alignment score significantly increases the performance for NDCG metric, we perform four A/B Tests using different parameter configurations when combining the two scores (i.e., the parameter deciding the proportion between *Similar Profiles* score and sequence alignment score used in the bagging scheme). We ran the AB test for one month. Results of *Profile Views* and *InMails* are summarized in Table 1. We denote the base *Similar Profiles* score by *sp* and sequence alignment scores derived by *SimCareers* by *sc*. As discussed in section 3.5.2, we do a linear combination of *sp* and *sc* score via parameter δ i.e. unified relevance score = $sp + \delta * sc$.

Table 1: The results of A/B Tests under different parameter configurations: the number in the table indicates the percentage increase (+) or decrease (-) in metrics using *SimCareers* compared with the baseline – *Similar Profiles*.

δ	Profile Views	InMails
0.01	-1.52%	$+0.83\%$
0.10	-10.33%	-7.51%
1.00	$\mathbf{+21.14\%}$	$+8.77\%$
10.0	$\mathbf{+20.12\%}$	$\mathbf{+12.11\%}$

Table.1 demonstrates the impact of A/B test results on our business metrics namely, *Profile Views* and *InMails*. These results further validate our hypothesis that capturing temporal information of career trajectory is crucial in computing profile similarity for hiring. More specifically, the δ value indicates the weight of the *SimCareers* score in the final unified relevance score. The results indicate that when the δ value is small, the difference between *SimCareers* score and *SimProfiles* score is negligible. It is because *sp* score dominates the *SimCareers* results with a small δ value. However, as we give more weight to the *SimCareers* score, we get a larger increase in both *Profile Views* and *InMails* sent. It is because that sequence alignment methodology captures the career trajectory information and thus can better measure the similarity between two profiles.

5. RELATED WORK

In this paper, we design a framework to model the similarity over a professional network. Although there are many studies on modeling similarity on social networks, none of them have taken temporal information into account in their models.

Similarity modeling over social networks is a hot topic, especially for online social networks such as Facebook, LinkedIn, and Twitter. A few traditional metric can be used to model similarity based on the network structure information, such as common neighbor (CN) [14] and Adamic-Adar (AA) [1]. Note that these methods are only based on network structure. Recently, a few algorithms have been proposed to model similarity using user profile information on social networks. The most relevant one along this line was done by Centinatas et al. [4]. This work aimed to model similarity for professional networks by extracting keywords from member profiles, and then treating keywords as features to train a discriminative model. This work, as well, only considers keyword based similarity. There are also a few similarity modeling methods [7, 11] that are proposed for other online social networks. All these profile based algorithms only use keywords extracted from user profiles and fail to consider temporal information of keywords.

Our paper also relates to sequence and time series data similarity [16, 6]. During past few years, due to its effectiveness, time series data similarity modeling have been widely applied in a many application domains. For instance, sequence alignment is used for detecting similarity between DNA, RNA or protein sequences [16]. In speech recognition, a few algorithms such as Dynamic Time Warping [20], have been proposed to measure similarity between audio sequences. Reyes et al. [18] applied time series data similarity in gesture recognition and achieved nice accuracy. He [9] used

sequence alignment to learn features for natural language processing. Yamano et al. [24] employed sequence alignment algorithm to analyze financial data. Although time series similarity is well-established problem in various fields, in this paper it is the first trial to apply sequence alignment to a field where each sequence is used to model the professional career in professional social networks. Moreover, different from many state-of-the-art methods, not only first-order features (i.e., position features) but also second-order features (i.e., transition features) are extracted from nodes of each sequence when training the node level similarity.

6. CONCLUSION AND FUTURE WORK

In this paper, we propose an approach *SimCareers*, to model member similarity over professional networks. In *SimCareers*, the member profile is modeled as a sequence of work experiences, while the career sequence similarity is evaluated using sequence alignment. To the best of our knowledge, *SimCareers* is the first similarity learning framework over online social networks that considers the career trajectory information. We use both offline and online experiments to demonstrate the effectiveness of *SimCareers*.

We believe *SimCareers* can be further refined in a variety of ways. Firstly, we intend to strengthen node-level similarity measure by incorporating per position geo information. In our existing profile dataset, we only have geo information with regards to member's current position. In the baseline keyword-based *Similar Profiles* system, geo is one of the important signals so we believe having per-position geo information would greatly improve *SimCareers*. Secondly, while aligning sequences in addition to skipping and matching the nodes we intend to allow the possibility of merging two similar nodes into one. A large number of career transitions are either lateral moves where the member changes company but does similar work at the same/similar role. Or they are promotions where the broad responsibility remains the same with a minor change in title. In both the scenarios, the core role & responsibility of the individual remains roughly the same. The hypothesis is that such a merge will help make the similarity measure more robust w.r.t. differences in how people fill their LinkedIn profile. Last but not least, while we are using the features listed in Section 3.3.2 to model the career positions, techniques such as feature selection [23] or feature interaction could be employed to explore better solutions at the feature level.

Furthermore, we wish to leverage *SimCareers* to help professionals do career planning. More concretely, by comparing career paths between young professionals and the early stage profiles from those of more successful senior individuals, we can give people a preview of possible career trajectories based on where they are currently. Furthermore, we can help young professionals decide which schools to choose, area of specialization to pursue and skills to acquire based on their desired success criterion as reified by an existing LinkedIn profile that these professionals wish to have as a role model.

7. REFERENCES

[1] L. A. Adamic and E. Adar. Friends and neighbors on the web. *Social Networks*, 25(3):211–230, 2003.

[2] K. Bartz, V. Murthi, and S. Sebastian. Logistic regression and collaborative filtering for sponsored search term recommendation. In *EC*, 2006.

[3] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, NY, 2006.

[4] S. Cetintas, M. Rogati, L. Si, and Y. Fang. Identifying similar people in professional social networks with discriminative probabilistic models. In *SIGIR*, 2011.

[5] D. Chakrabarti, D. Agarwal, and V. Josifovski. Contextual advertising by combining relevance with click feedback. In *WWW*, pages 417–426, 2008.

[6] Z. Chen. Mining individual behavior pattern based on significant locations and spatial trajectories. In *PerCom*, 2012.

[7] J. Golbeck. Trust and nuanced profile similarity in online social networks. *ACM Trans. Web*, 3(4):1–33, 2009.

[8] A. K. Hartmann. Sampling rare events: Statistics of local sequence alignments. *Physical Review E*, 65(5), 2002.

[9] J. He. Improving sequence alignment based gene functional annotation with natural language processing and associative clustering. In *ISNN*, 2010.

[10] K. Jarvelin and J. Kekalainen. Cumulated gain-based evaluation of ir techniques. *TIST*, 20(4):422–446, 2002.

[11] A. Jeckmans, Q. Tang, and P. Hartel. Privacy-preserving profile similarity computation in online social networks. In *ACM conference on Computer and communications security*, pages 793–796, 2011.

[12] R. Kohavi, R. Longbotham, D. Sommerfield, and R. M. Henne. Controlled experiments on the web: survey and practical guide. *Data Mining and Knowledge Discovery*, 18(1):140–181, 2009.

[13] D. Kreider, D. Lahr, and S. Diesel. *Principles of Calculus Modeling: An Interactive Approach*. Springer, NY, 2005.

[14] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM*, 2003.

[15] E. Montanes, J. R. Quevedo, I. Diaz, and J. Ranilla. Collaborative tag recommendation system based on logistic regression. In *ECML*, 2009.

[16] D. Mount. *Bioinformatics: Sequence and Genome Analysis*. Cold Spring Harbor, NY, 2004.

[17] A. Reda, Y. Park, M. Tiwari, C. Posse, and S. Shah. Metaphor: a system for related search recommendations. In *CIKM*, pages 664–673, 2012.

[18] M. Reyes, G. Dominguez, and S. Escalera. Featureweighting in dynamic timewarping for gesture recognition in depth data. In *ICCV*, 2011.

[19] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: Estimating the click-through rate for new ads. In *WWW*, pages 521–529, 2007.

[20] S. Salvador and P. Chan. Toward accurate dynamic time warping in linear time and space. In *KDD Workshop*, 2004.

[21] `http://www.forbes.com/sites/georgeanders/2013/04/10/whoshould-you-hire-linkedin/-says-try-our-algorithm/`.

[22] M. Vingron and M. S. Waterman. Sequence alignment and penalty choice: Review of concepts, case studies and implications. *J. Molecular Biology*, 235(1):1–12, 1994.

[23] Y. Xu and D. Rockmore. Feature selection for link prediction. In *PIKM*, 2012.

[24] T. Yamano, K. Sato, T. Kaizoji, J.-M. Rost, and L. Pichi. Symbolic analysis of indicator time series by quantitative sequence alignment. *Journal of Computational Statistics and Data Analysis*, 53(2):486–495, 2008.

[25] Z.-H. Zhou. *Ensemble Methods: Foundations and Algorithms*. Chapman and Hall, 2012.