Social Skill Validation at LinkedIn

Xiao Yan Jaewon Yang Mikhail Obukhov Lin Zhu Joey Bai Shiqi Wu Qi He
LinkedIn
Mountain View, California
{xyan,jeyang,mobukhov,lizhu,jobai,sqwu,qhe}@linkedin.com

ABSTRACT

The main mission of LinkedIn is to connect 610M+ members to the right opportunities. To find the right opportunities, LinkedIn needs to understand each member's skill set and their expertise levels accurately. However, estimating members' skill expertise is challenging due to lack of ground-truth. So far, the industry relied on either hand-created small scale data, or large scale social gestures containing a lot of social bias (*e.g.*, endorsements).

In this paper, we develop the *Social Skill Validation*, a novel framework of collecting validations for members' skill expertise at the scale of billions of member-skill pairs. Unlike social gestures, we collect signals in an anonymous way to ensure objectiveness. We also develop a machine learning model to make smart suggestions to collect validations more efficiently.

With the social skill validation data, we discover the insights on how people evaluate other people in professional social networks. For example, we find that the members with higher seniority do not necessarily get positive evaluations compared to more junior members. We evaluate the value of social skill validation data on predicting who is hired for a job requiring a certain skill, and model using social skill validation outperforms the state-of-the art methods on skill expertise estimation by 10%. Our experiments show that the *Social Skill Validation* we built provides a novel way to estimate the members' skill expertise accurately at large scale and offers a benchmark to validate social theories on peer evaluation.

CCS CONCEPTS

• **Computing methodologies** → **Classification and regression trees**; • **Information systems** → **Social networks**.

KEYWORDS

skill validation, social signals, behavior pattern

ACM Reference Format:
Xiao Yan Jaewon Yang Mikhail Obukhov Lin Zhu Joey Bai Shiqi Wu Qi He. 2019. Social Skill Validation at LinkedIn. In *The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19), August 4–8, 2019, Anchorage, AK, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3292500.3330752

KDD '19, August 4–8, 2019, Anchorage, AK, USA

ACM ISBN 978-1-4503-6201-6/19/08…$15.00
https://doi.org/10.1145/3292500.3330752

1 INTRODUCTION

The LinkedIn professional social networks serve 610M+ professionals in the world [1]. LinkedIn members use LinkedIn to manage and build their professional careers: managing their professional profiles, organizing their connections, learning new courses, reading articles, browsing job postings and searching for other people and so on.

LinkedIn allows members to list their skills on their profile [8, 15]. Each skill is a keyword that represents a professional topic. Examples of skills include tools such as "MATLAB" or "Python"; They also include industrial knowledge such as "Machine Learning" or "Artificial Intelligence" or soft skills such as "Presentation" or "Technical Writing". Skills are used in many applications to target the right audience. For example, advertisers show their ads to members with a certain skill and recruiters use skills to query right candidates to reach out.

This paper aims to tackle a problem of identifying people who have expertise in a certain skill [2, 4, 15]. The skills in the LinkedIn members' profiles may be inaccurate or outdated. LinkedIn members may list any skill they want on their profile and sometimes they add skills that they do not really know about. In this paper, we develop a method to identify what skills each member has expertise in.

Identifying the skill expertise of members will remove the noise in LinkedIn's skill data, and thus improve the quality of LinkedIn products that rely on skills. For example, recruiters can reach out to candidates who have expertise in certain skills (not those who claim to have the skills) [15]. It also helps the profile owners as they can tell what skills they have expertise and what skills they do not (and need to improve) [8].

The problem of identifying the skill expertise has been studied as finding experts in a certain topic (skill) in the research community [2, 4, 15], but it remains as a very hard problem. The main challenge is the lack of large scale ground-truth [8]. It is an expensive task to assess whether a person has an expertise in a certain skill because it requires good understanding of the person and the skill.

In this paper, we aim to find a solution that satisfies the following properties. First, we want to estimate the members' expertise in an objective and unbiased way. If our estimates are biased towards a certain kinds of users (e.g., very active users), our estimates will not help other LinkedIn products. Second, we want to identify members' skill expertise across all skills at LinkedIn. As mentioned above, LinkedIn skills include various kinds such as tools, industrial knowledge and soft skills, and we want to cover all kinds of skills. Third, we aim to build a sustainable framework where we can evaluate and improve our methods over time continuously.

There are existing methods to identify skill expertise, but we argue that these methods do not satisfy the three conditions mentioned above. One line of research is developing unsupervised methods such as PageRank [27]. The main drawback with these unsupervised methods is that it is affected by certain user behavior. For example, if we run PageRank on member connection data, the output heavily depends on member connections and will favor people who tend to connect to high-profile users. Another drawback is that it is hard to evaluate the models (due to lack of supervised data) and further improve the models.

An alternative to unsupervised methods is to train a supervised model based on a small set of hand-curated data [15]. LinkedIn developed "Skill Reputation" based on an external dataset (e.g., a list of Apache committers). While this method outperformed unsupervised methods in the experiments, this method has a drawback that such external gold-standard data set is concentrated on a certain kinds of skills. For example, Apache committers cover only programming-related skills.

Another way of solving this problem is to collect signals from users [8]. This way, we can collect signals for various kinds of skills. Also, we can collect large scale data with which we can evaluate and improve our models systematically. With these benefits in mind, LinkedIn introduced "Endorsements" where we ask members to endorse other users' skills. The drawback with this approach is that endorsements are visible to all users and became social gesture rather than objective assessment of skill expertise [8]. Another drawback of endorsements is that it is not discriminative enough; we observe only positive feedback and no negative feedback.

In this paper, we propose building a framework to estimate members' skill expertise. In particular, we present the *Social Skill Validation* product to collect the assessment of members' skill expertise from other members. Through this product, we can collect ground-truth signals on members' skill expertise and can address the major challenge in lack of ground-truth. We ask members to assess other members' skill levels in three different ways that we explain later.

We argue that our solution satisfies the three conditions we mentioned: Getting objective estimates, achieving good skill coverage and building sustainable framework. Like endorsements, we collect member feedback continuously and thus achieve good skill coverage and build a sustainable framework. Unlike endorsements, however, members' assessments are anonymous, and therefore provides better objectivity. We also make sure our questionnaires have discriminative nature; we ask members to rank other members with respect to a certain skill.

Using data collected from *Social Skill Validation* product, we can also learn about how people evaluate other people. For example, do people become more selective when they evaluate others with a similar status as them [3]? Or do people tend to get favorable feedback as they get more senior? We can validate these hypotheses with the data we collect.

This paper makes the following contributions in the problem of estimating members' skill expertise:

- We present *Social Skill Validation*, the first framework to collect objective, discriminative feedback on professional social networks.
- We perform rigorous data analysis to understand patterns in how people assess other peoples' skill expertise. We check whether our data contains bias and noise from social aspect and propose ways to address them.
- Based on the patterns we learned from the analysis, we further improve the product by developing a machine learning model to make better suggestions to the evaluators.
- We verify the quality of the feedback we received. We test the usefulness of the feedback on an separate gold-standard dataset that was manually created in an independent way. In the experiments, social skill validation signals help improving the classification accuracy by 10.3%.

2 PROBLEM DEFINITION

Social Skill Validation is aiming to solve the following problem: Given the set of skills member claim to have on the profile page, find a subset of skills that the member truly has expertise in. The main obstacle of solving this problem is the lack of ground-truth data. In this paper, we propose a solution to this problem, by virtue of using social signals to predict validation label of skills.

Definition 2.1. **Social Signal** is a recorded action that member a performs with regard to member b, $a, b \in M$. M is the collection of members in LinkedIn's 610M+ social network. In our case, a social signal is the assessment of a on whether b has the skill s.

Definition 2.2. **Skill Set** S^* of a member m is a subset of skills, that member claimed to have by putting them in the *Skill* section of member profile. $S^* \subset S$, S is the collection of skills defined in LinkedIn taxonomy. At LinkedIn we allow a maximium of 30 skills per member.

Definition 2.3. **Validation Label** $y_l \in \{0, 1\}$ is a binary label whether a member m's expertise on skill s is validated (i.e., the member truly has the skill).

Definition 2.4. **Skill Validation** is the task of finding the function f, that $y_l = f(m, s; \theta)$, for member m and skill s, $s \in S^*$, θ is the collection of input features.

In this paper, we propose a novel method of using social signals to generate θ, and showcase the efficiency and accuracy of our method with a separate gold-standard dataset.

3 METHODS

3.1 Dataset Collection

The core of *Social Skill Validation* is the generation of social signals. Assessment from member him/herself is not reliable because member tends to be overconfident about the expertise of his/her own skills. On the other hand, assessments from skill experts or recruiters are objective, but they do not have first hand experience with the candidate, and are easily biased towards the description on the member profile. We choose to collect skill assessments from member's connections to achieve both objectivity and accuracy. At the first stage, we launch three different promos on the member profile page of LinkedIn, each targeting a different aspect of skill assessment.

Go-to-Connection is used to collect viewer's rank on candidates' expertise on a particular skill. This promo is triggered when

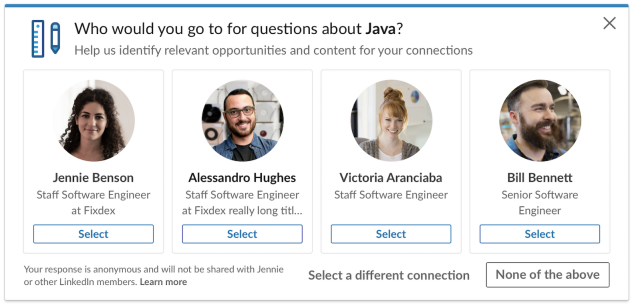

Figure 1: A Go-to-Connection promo showing the four candidates.

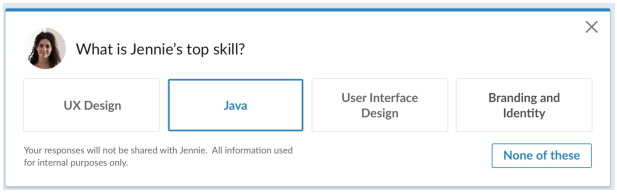

Figure 2: A Top Skill promo showing the four skills of the candidate.

a user is viewing a profile page of another user. This promo will give a list of four candidates, and ask the viewer to select the expert of the skill among the four choices. Choices such as choosing a candidate beyond the four, and choosing "none-of-the-above" are also given to the user for a fair assessment (Figure 1). The choices are referred to as *Position* $0 \sim 3$ in the following text. It is worth pointing out that the *Position 0* is always the viewee whose profile is viewed by the current viewer. And *Position* $1 \sim 3$ are randomly selected from the first-degree connections of the viewer.

Top Skill makes a member compare a connection's skills within a certain category (Industry Knowledge, Interpersonal Skills, Tools & Technologies). It is triggered when visiting the personal page of the candidate. The viewer is asked to choose one skill out of 4 potential choices as the candidate's top skill in that category (Figure 2).

Endorsement FollowUp Endorsement is one of LinkedIn's key feature as a career network, that it asks the viewer to endorse a skill for the candidate, as a proof of the candidate's skill level [8]. Endorsement FollowUp is built upon LinkedIn's unique endorsement feature, that it is triggered by the endorsement event. After viewer endorses the candidate for a given skill, additional promo will show up to ask the viewer the skill expertise of the candidate on the skill endorsed, and the relationship between viewer and the candidate. This is the only signal we collected that also takes viewer-candidate relationship into consideration (Figure 3).

All promo results are invisible to the candidates for an objective evaluation. 6 months of data is collected as viewer v, candidate c, skill s and label $y \in \{0, 1\}$ (viewer chooses the candidate as the skill expert in the current promo session, or in the Endorsement FollowUp case, viewer selects "highly skilled"). Relative numbers of data collected in each signal are shown in Table 1.

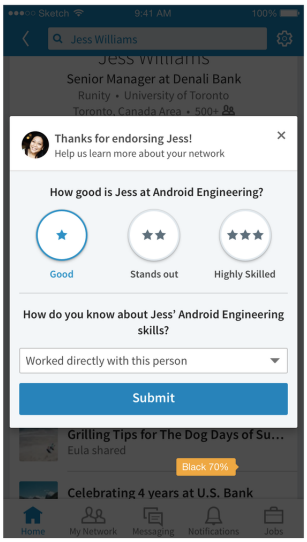

Figure 3: An Endorsement FollowUp promo showing the options.

Table 1: Social Validation Datasets

Datasets	Relative Counts
Endorsement FollowUp	100%
Go-to-Connection	41.4%
Top Skill	68.5%

3.2 Machine Learning Based Data Collection Method for Go-to-Connection

In Table 1, Go-to-Connection is the weakest signal at this moment due to low user participation. This signal is very critical for us, because this is the only signal that directly ranks the candidates for a given skill. The signal of this promo will decide the relative skill expertise among members.

We decide to use machine learning models to optimize the data collection process, i.e., by increasing the user response rate. This problem can be defined as follows: a viewer v needs to choose the expert e from a set of candidate C for a given skill s . We model this problem as finding the candidate $c \in C$ that maximizes the probability of being chosen by v as e, i.e.:

$$c = \underset{c}{\text{argmax}} \Pr(e = c \mid c, v, s), c \in C$$

3.2.1 Modeling. Based on the setup of the promo, a random selection (where for each $\langle v, c, s \rangle$, a random score y_r between $[0, 1]$ is given) is used as the baseline to simulate the current implementation.

We use features including *Social Status*, *Skill Reputation*, *Viewer-Candidate Relation* to utilize the user behavior pattern and compensate for signal-specific bias (detailed description and analysis of the features can be found in Section 4). Another feature is a LinkedIn member profile-specific feature, *Skill Rank*, the rank of a

skill appearing in the *Skill* session of a member profile. This feature is used to represent the relative expertise of a skill compared to other skills in S^* from member's own judgement.

For modeling the problem, we use a linear logistic regression model and non-linear gradient boosted tree model (XGBoost [9]). In logistic regression, for each feature, we add one indicator to indicate the existence of this feature.

For each promo session, data involving candidate at *Position 0* is removed. By design *Position 0* always shows the profile viewee, and cannot be changed by our model. Position bias is reported in previous research that option appearing in a higher position is more likely to be chosen by user [14, 26]. Candidate position is added during training to capture the position bias but removed from validation and test stages. Candidate position is not used in XGBoost model due to the fact that removing a feature from a tree completely destroys the tree structure below the feature node.

3.2.2 Evaluation. To evaluate the effectiveness of the model, we choose AUC (area under ROC curve), measuring the probability of ranking a random positive sample ahead of a negative sample.

Another evaluation method is the Mean Average Precision (MAP). For each promo session, we calculate the Average Precision (AP) @1, which in our case, is the indicator whether expert e viewer v chooses among a set of candidates C is the candidate that has the highest score based on our model. Then we average AP@1 for all sessions to obtain the MAP@1.

$$\text{AP}\ @K = \frac{\sum_{i=1}^{K} \text{P}\ @i}{K}$$

$$\text{MAP} = \frac{\sum_{q=1}^{Q} \text{AP}(q)}{Q}$$

In the above definition, P is precision, K is the cutoff rank of the top most results when calculating AP. In our case, $K = 1$, and Q is the number of sessions.

3.3 Skill Validation Modeling Using Social Signals

So far, we describe how we collect the data to tackle the main problem that we defined in Section 1. In the previous sections, through three different methods, we collect viewer assessments of the candidate $\langle member, skill \rangle$ pairs. In this section, we tackle the main problem in Section 1 by combining the signals we collected.

3.3.1 Golden Dataset. We frame this method as a supervised learning problem. In order to generate ground truth data, we leverage members' behavior and profile update data to generate a dataset named *Confirmed Hire*, which has the following definition: a member viewed or applied for a job on LinkedIn, and got hired within 6 months. Using *Confirmed Hire* data, we align the skills that member claims to have with the skills required by job posting, and use the result $\langle member, skill \rangle$ pairs as the ground truth. *Confirmed Hire* only covers a very small percentage of members, and on average < 2 skills per member. Due to the small scale of the data set, we choose to use it as a golden data set instead of a solution to the skill validation problem.

3.3.2 Feature Engineering. The main challenge of using multiple social signals is that, due to the design difference, each obtained signal does not necessarily agree with each other. Chen 2017 [8] proposed a modeling-based approach to develop quality metrics from less reliable signals. In another study, Ratner 2017 [22] used a generative model to generate final label with a list of weak labeling functions. We adopt the concepts described in [22] and [8], and develop our own method for utilizing each social signal.

For each member m, we generate a feature vector $\phi(\cdot)$ as follows:

$$\phi_{ij}^{sig}(\mathbf{W}) = w_{ij}$$
$$\phi_{ij}^{cnt}(\mathbf{W}) = count(w_{ij})$$
$$\phi_{ij}^{sig\_lab}(\mathbf{W}) = \mathbb{1}\{w_{ij} \neq \varnothing\}$$
$$\phi_{ij}^{cnt\_lab}(\mathbf{W}) = \mathbb{1}\{count(w_{ij}) > 1\}$$
$$\phi_{ijl}^{corr}(\mathbf{W}) = \mathbb{1}\{w_{ij} = w_{il}\}, j, l < k, j \neq l$$
$$\phi_{i}^{maj\_vote}(\mathbf{W}) = majority\_vote(w_{i0}, w_{i1}, ..., w_{ik})$$

We model each signal as a noisy "voter" $\mathbf{w}$. For each member m and skill set S^*, we collect the signal matrix $\mathbf{W} = \{\mathbf{w_0}, \mathbf{w_1}, ..., \mathbf{w_k}\}$, while $\mathbf{w_j} = \{w_{0j}, w_{1j}, ..., w_{nj}\}$. w_{ij} is the signal for the ith skill from jth voter, k is the number of signals/voters and n is the length of S^*. $count(\mathbf{w})$ is a function returns the number of votes for voter $\mathbf{w}$. For each voter $\mathbf{w}$ and $count(\mathbf{w})$, we generate propensity features (ϕ^{sig_lab} and ϕ^{cnt_lab}) to compensate for the missing and strength of the votes. To model the statistical dependencies between the voters, we also add features representing pairwise correlations (ϕ^{corr}). Finally, we calculate the overall trend of the voters, using a function $majority\_vote(\mathbf{w_0}, \mathbf{w_1}, ..., \mathbf{w_k})$ that takes simple majority vote over the k voters.

3.3.3 Modeling and Evaluation. As a baseline, we use the *Skill Reputation* model (Ha-Thuc 2016) [15] to predict validation label of $\langle member, skill \rangle$ pairs, because this is the current standard LinkedIn uses. Another baseline we use is a metric developed by Chen 2017 [8], which records number of endorsements for $\langle member, skill \rangle$ pairs. We normalize the endorsements count to [0, 1] as the prediction result. In our new method, features extracted from each signal are incorporated by stacking a XGBoost model on top of the method described in Ha-Thuc 2016. We choose non-parametric tree models for the purpose of handling categorical features and capturing unobserved interactions. AUC and accuracy are used as metrics to measure model performance.

4 PATTERN OF MEMBER BEHAVIOR IN SKILL ASSESSMENTS

In this section, we analyze the three datasets and discover the patterns in each dataset. These patterns will not only help us identify potential bias in assessments, but also verify hypotheses on member behaviors. Specifically, we study the effect of the *Social Status*, *Skill Reputation*, and *Viewer-Candidate Relationship* on the three datasets.

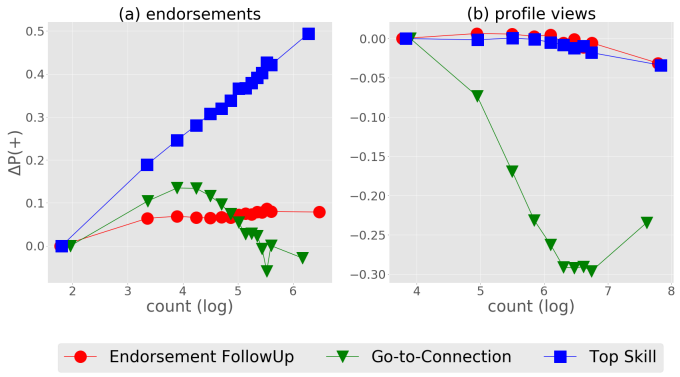

Figure 4: Change of *P(+)* ($\Delta P(+)$) as a function of *Social Status*. *P(+)* is normalized to the first data point to show the $\Delta P(+)$ with *Social Status*

4.1 Social Status and Evaluations

Social Status is defined as a function of social recognition of a member's activity on a social network. We adopt previous research on using volumetric measures as approximations of status [3]. We use two aggregated metrics LinkedIn uniquely possesses as a professional network: number of endorsements (D), and number of profile views (R), with the following definition:

$$D_c = \sum_{v \in M, v \neq c} \sum_{s \in S} is\_endorse(v, c, s), c \in M$$

$$R_c = \sum_{v \in M, v \neq c} \sum_{t \in [t_s, t_e]} is\_profile\_view(v, c, t), c \in M$$

M is the collection of members, S is the collection of skills at LinkedIn. *is_endorse* is an indicator function that returns whether viewer v endorses candidate c for skill s. *is_profile_view* is an indicator function that returns whether v visits c's profile page at time t. t_s, t_e are the start and end points of the aggregated time period.

A natural assumption is that *Social Status* is positively correlated with favorable response in skill assessments. We verify this hypothesis with the collected data. In Figure 4 we first plot positive response rate (*P(+)*, the fraction of positive response in a given set of assessments [3]) as a function of *Social Status*. However, Go-to-Connection shows a dramatic decline in *P(+)* when endorsements count is higher than a boundary point. In Endorsement FollowUp and Top Skill, *P(+)*s are insensitive with the number of profile views, which represents the fact that profile views is a much more heterogeneous metric. To our surprise, *Go-to-Connection* shows a clear negative correlation between *P(+)* and profile views.

To understand the reason of the *P(+)* decline in Go-to-Connection, we study the correlation between *Social Status* and member characteristics. One hypothesis is that the seniority level of candidate, which generally increases with the *Social Status*, is influencing viewer's choice. To represent the seniority level of member, we utilize a numerical system LinkedIn developed to determine the seniority level of a member, the seniorityId. SeniorityId is in the scale of 0 ~ 10, with 10 being a most senior position (founder, CEO,

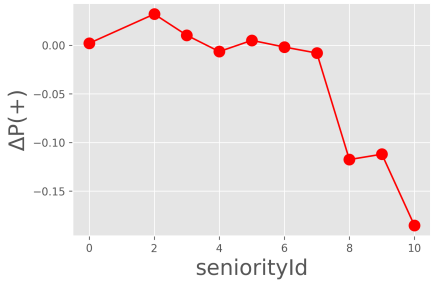

Figure 5: $\Delta P(+)$ as a function of seniorityId in *Go-to-Connection*. *P(+)* is normalized to the first data point to show $\Delta P(+)$ with seniority

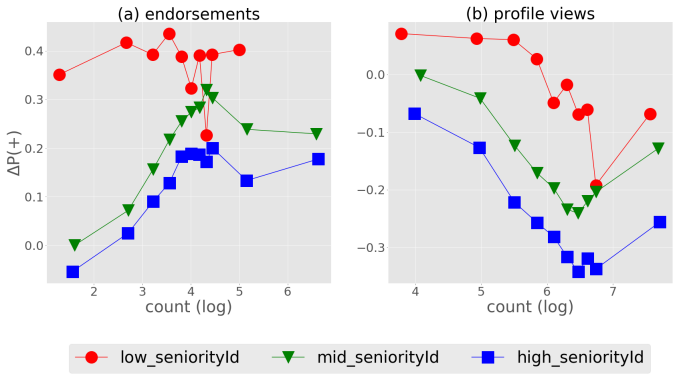

Figure 6: $\Delta P(+)$ as a function of *Social Status*, with respect to different seniority groups. *P(+)* is normalized to the first data point of mid-seniorityId group to show $\Delta P(+)$ with *Social Status*

president, etc) and 0 being unoccupied. In Go-to-Connection, the seniority of the candidate is negatively correlated with *P(+)* (Figure 5). One possible explanation is that, people tend to be cautious to select a high-seniority connection that is closely related to themselves, such as managers, advisers, etc. However, we cannot rule out the possibility that because we are asking the viewer "the one to ask questions", not necessarily interpreted as "the one with the best skills", it could be simply a side effect of the product design. Candidates are divided based on seniority: high, mid and low seniority groups, and *P(+)* is plotted as a function of *Social Status*. Low-seniority group show no correlation between endorsements count and *P(+)*, while in mid and high seniority groups *P(+)* increases with number of endorsements (Figure 6). This data indicates that endorsement signal is more reliable when seniority level passes certain threshold, while for junior members the endorsements are more of a social gesture.

4.2 Skill Reputation and Evaluations

LinkedIn developed a state-of-the-art machine learning model to rank the skill expertise of $\langle member, skill \rangle$ pairs for LinkedIn's skill taxonomy [15], referred to as **Skill Reputation**. *Skill Reputation* is defined as an expertise score for both skills that members list

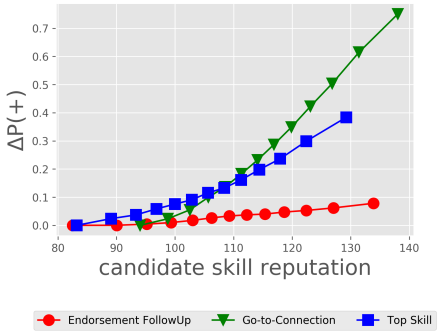

Figure 7: Δ*P(+)* as a function of candidate skill reputation. *P(+)* is normalized to the first data point to show the Δ*P(+)* with candidate skill reputation

on their profiles as well as the skills that members potentially possess [15]. To study the relationship between skill expertise and assessments, we utilize the *Skill Reputation* score as an estimation of member's skill expertise.

P(+) grows with the *Skill Reputation* of the candidate in all datasets (Figure 7). This is expected because the task of the promos is to find the expert for a certain skill. However, different promos show different sensitivity with respect to *Skill Reputation*, suggesting that different aspects of information is collected in each promo.

Previous research [3] shows that the difference of status between viewer and candidate is the primary cause of the change in *P(+)*. To test this hypothesis, in Figure 8 we plot the Δ*Skill Reputation* between viewer and candidate (viewer *Skill Reputation* - candidate *Skill Reputation*). We divide the viewer into three groups based on the *Skill Reputation*: high skill viewer, mid skill viewer and low skill viewer to assess the effect of viewer *Skill Reputation* on the response. Plot with *P(+)* shows that regardless of the viewer skill reputation, in most cases the Δ*Skill Reputation* negatively correlates with the *P(+)* in all three datasets. Viewer is more willing to offer positive response when he/she is at a much lower *Skill Reputation* than the candidate, but less likely to give a positive response when the viewer's skill reputation is higher than the candidate. The only exception is the low skill viewer group in Endorsement FollowUp. We believe this exception is caused by the incompetence of low score viewer on the task of assessing others' skills, that extra caution should be taken when handling the Endorsement FollowUp signals from low skill viewer. At the same level of Δ*Skill Reputation*, high skill reputation viewers give higher *P(+)* than mid or low skill viewers. In the case of Go-to-Connection, the sensitivity of viewer response on the Δ*Skill Reputation* increases with the viewer skill reputation, as shown by the slope change in Figure 8(b). This data, as the first recorded evidence, indicates that the confidence of viewer increases with his/her own skill expertise, and the high skill viewer is more willing to respond based on the skill expertise difference.

4.3 Viewer-Candidate Relation and Evaluations

Viewer-Candidate Relation is defined as a function of the social interactions and connection strength between viewer node v and candidate node c in LinkedIn's social network graph. Based on the product design, the viewer is asked to assess the skill expertise of his/her first degree connections. We hypothesize that the relationship between the viewer and candidate will play a positive role for the assessment decision.

To verify our hypothesis, we use two numerical metrics as approximation of *Viewer-Candidate Relation*: the number of viewer-candidate profile views P^* that v gives c, which is single directional metric.

$$P^*_{vc} = \sum_{t \in [t_s, t_e]} is\_profile\_view(v, c, t), v \in M, c \in M$$

The second one is *People You May Know* (PYMK) score developed by LinkedIn. Each ⟨*member a*, *member b*⟩ pair is given a score $\in [0, 1]$ by the model, which is a bi-directional, quantitative representation of how close two members are [17, 18]. Figure 9 reveals a strong positive correlation between the number of viewer-candidate profile views and *P(+)* in Go-to-Connection, while in Top Skill and Endorsement FollowUp, relationship between related profile views and *P(+)* is not so obvious.

Similarly, *P(+)* increases with the PYMK Score between viewer and candidate in Go-to-Connection, which is a strong evidence that viewer will prefer the candidate that has a closer relationship. The same phenomenon is not observed in Top Skill and Endorsement FollowUp. We believe this is caused by the design decision that in Go-to-Connection, the candidate is not fixed as in the cases of Top Skill and Endorsement FollowUp, that viewer is given the freedom of choosing from a list of candidates. This again reinstates the importance of collecting multiple signals for the assessment task.

Table 2: Models of Go-to-Connection Recommendation

Model	AUC	MAP@1
Baseline	-	-
Logistic Regression	+26.4%	+48.5%
XGBoost	+30.8%	+54.5%

5 EXPERIMENTS

5.1 Performance of Go-to-Connection Model

In this subsection, both offline and online results of the Go-to-Connection data collection model described in Section 3.2 are shown below.

5.1.1 Offline. Offline result is shown in Table 2. Due to company policy, only the relative change compared to baseline is shown. Both logistic regression and XGBoost model lead to a large lift on AUC and MAP@1. The performances of the models are similar, with XGBoost performs slightly better.

5.1.2 Online. We decide to launch the logistic regression model online over XGBoost for the following reasons: logistic regression clearly removes the position bias, and the in house implementation of logistic regression is more convenient and robust for online serving purposes. When a viewer is asked to choose an expert for a skill, all possible candidates are scored by the model, ranked

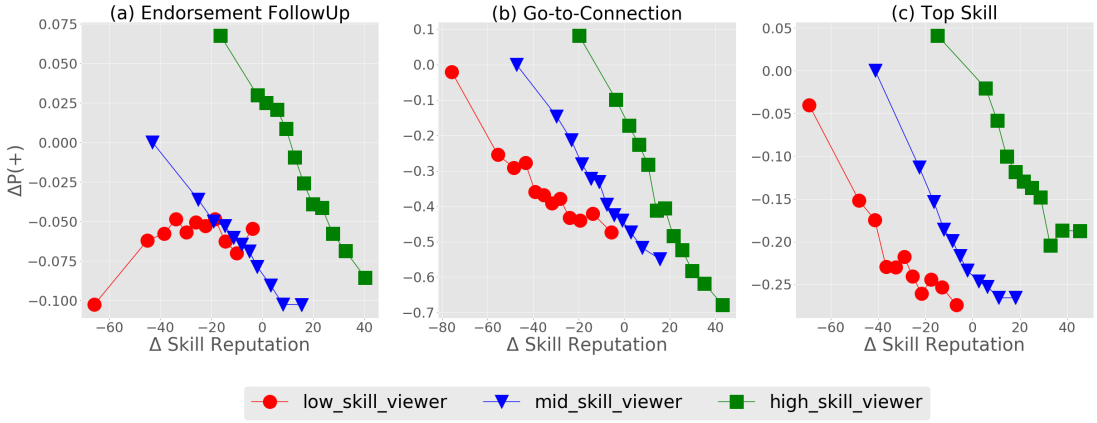

Figure 8: $\Delta P(+)$ as a function of Δ*Skill Reputation*, with respect to different viewer groups. *P(+)* is normalized to the first data point of mid skill viewer to show the $\Delta P(+)$ with respect to Δ*Skill Reputation*

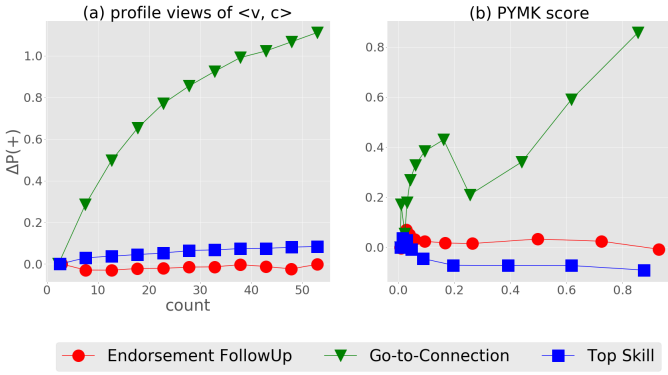

Figure 9: $\Delta P(+)$ as a function of *Viewer-Candidate Relation*. *P(+)* is normalized to the first data point to show the $\Delta P(+)$ with *Viewer-Candidate Relation*

based on the model score, and top 3 candidates are shown to the viewer together with the viewee whose profile page is viewed by the viewer. A/B test is performed by splitting LinkedIn members in 50/50 ratio. Test group served by our model have a 4.0% significant increase in number of viewers selecting from the promo, while a 39% decrease in the number of viewers skipping the promo (Table 3). This data indicates that our model significantly improves the user response rate of the promo.

Based on the current product design, viewer can be asked to choose a skill expert for the same skill multiple times, with different sets of candidates. To see the effect of the model in situations where viewer has already chosen a skill expert, we divide the test data into viewer with and without a previously selected skill expert for a certain skill. With our model, viewer selections increase 6.5% for choosing a skill expert for the first time. However, there is a dramatic 23.1% drop in viewer selections of a different skill expert than their previous choice (Table 3). These results imply that our model generates more reliable candidate, such that the users tend to stay with their choices under our model. The A/B test results show

Table 3: A/B Test of Go-to-Connection Model Compared to Baseline

Number of	Change
Users selecting from promo	+4.0%
Users skipping the promo	-39.0%
User selections of a skill expert for the first time	+6.5%
User selections of a different skill expert than before	-23.1%

a strong evidence that our modeling approach is capable of greatly lifting not only the signal collection efficiency, but also accuracy of the signal.

5.2 Performance of Skill Validation Models with Social Signals

In this subsection we show the experiment results for model described Section 3.3.

Table 4 shows the result of adding additional signals, with comparison to *Skill Reputation* (Ha-Thuc 2016). The absolute numbers are hidden due to company policy. Endorsements count performs much worse than *Skill Reputation*, suggesting that the endorsements are not an accurate indicator for skill expertise due to social bias and noise. Adding all three signals outperforms both of the baselines, and has +10.3% lift in terms of AUC and a +7.7% lift on accuracy. This result proves the effectiveness of our method on collecting, utilizing, aggregating social signals, and improving the prediction efficacy.

To evaluate the relative contributions of signals in the model, we perform ablation study by removing each signal (Table 4). Of the three signals we asked for user input, removing Top Skill leads to the weakest lift in AUC and accuracy (+3.4% and +5.4% respectively). Removing Endorsement FollowUp shows similar effect as removing Go-to-Connection. However, none of them are comparable to adding all three signals together. This study demonstrates the necessity of applying multiple differently designed methods to collect signals.

Table 4: Skill Validation Models with Social Signals

Model	Accuracy	AUC
Ha-Thuc 2016 [15]	-	-
Endorsements Count	-3.5%	-4.1%
Ha-Thuc 2016 + 3 social signals	+7.7%	+10.3%
Ha-Thuc 2016 + Endorsement FollowUp + Top Skill	+5.1%	+7.0%
Ha-Thuc 2016 + Endorsement FollowUp + Go-To-Connection	+3.4%	+5.4%
Ha-Thuc 2016 + Top Skill + Go-To-Connection	+4.8%	+7.1%

Table 5: Model Lift Compared to Ha-Thuc 2016

Test Group	Accuracy	AUC
Programming skill	+9.5%	+2.7%
Non-programming skill	+7.4%	+9.9%

Skill Reputation model is trained with programming related skills (Apache commiters), which is potentially biased against non-programming skills. Indeed, the predictive power reduces > 10% for test data containing only non-programming skills, compared to test data containing only programming skills. Using our best model (Ha-Thuc 2016 + 3 social signals), we compute the lift compared to *Skill Reputation*, using test data containing programming skills or non-programming skills (Table 5). Our model leads to a comparable lift of accuracy in both test groups, and a much larger increase of AUC in test group with non-programming skills than with programming skills (AUC +9.9% vs 2.7%, respectively). This data indicates that a significant portion of the performance gain in our model is related to the better prediction with data that contains non-programming skills. It also suggests that our model generalizes better to skills of all fields.

6 RELATED WORK

6.1 Human Evaluations in Social Media

Evaluations in social media have several types [3]: 1) The evaluation of movies (Netflix), products (Ebay, Amazon) or websites (Google). In the evaluation process, only one person — the reviewer is involved [13, 16, 19, 23]. Methods to solve these problems normally fall into either collaborative filtering or recommendation systems. 2) Opinion mining and sentiment analysis is another large category, that by means of natural language processing (NLP), directly infers the option from natural-language text [10, 20]. And 3) the evaluation between two persons, either likes on a community platform such as Stack Overflow, or the promotion nomination of users as administrator on Wikipedia [2, 5]. Previous research shows that, similarity between the characteristics of two users, not the absolute status, is a more critical factor in affecting the evaluations between one another [3].

6.2 Modeling Viewer Response

The first problem we are facing when building the product is to make better suggestions for the evaluator to interact with. One related concept is the click model that predicts the click-through rate (CTR) of the user [7, 11]. In the development of the click model, position bias is observed that the option appearing in a higher position is more likely to be chosen by user [26]. Another way to formalize the problem is to treat it as a learning-to-rank problem, by predicting the ranks of the options with regards to viewer's preference [6]. There are three main approaches of ranking algorithm: the pointwise approach [12], the pairwise approach [21, 24], and the listwise approach [25]. The choice of the approaches depends on the purpose of the modeling, and the format of ground-truth data.

6.3 Combination of Multiple Signals for Reliable User Assessments

Another related area is the generation of reliable label with multiple weak signals. Ratner 2018 published on a novel method to denoise multiple labeling functions and generate the final label without manual evaluation [22]. The core of the framework is a generative model over the labeling functions by learning from the intersection and divergence of the labeling functions. Chen 2017 described a framework to develop metrics that defines valuable user evaluation [8]. In this paper, a broad set of relevant signals about the viewer and candidate, as well as the relationship between them were collected, and useful signals among them were then selected with a machine learning based approach.

7 CONCLUSIONS

We developed a novel framework, *Social Skill Validation* to identify LinkedIn user's skill expertise with social signals collected network wide. We have three ways of signal collection: Endorsement FollowUp is used to assess how strong the $\langle member, skill \rangle$ pair is after viewer endorses candidate on a skill, which removes weak endorsement signal; Top Skill asks the viewer to find most proficient skill among skills that the candidate claims to have, which eliminates the candidate overconfidence problem; Go-to-Connection is the sole signal that directly compares skill expertise among candidates, which is used to provide the expertise threshold for the validated skills. Analysis of the collected data reveals that, all three signals show similar patterns with *Skill Reputation*, while *Go-to-Connection* seems to be sensitive to seniority level of the candidate, as well as the connection strength between viewer and candidate.

In order to improve the signal collection efficiency in Go-to-Connection, we launched a logistic regression model, with features designed to utilize user behavior pattern and address the bias discovered during data analysis, and observed high lift on user response rate as well as data quality through online A/B test.

Finally we showcased the power and effectiveness of our approach, by extracting features from each signal and training a gradient boosted tree model. Our approach significantly outperformed the previous *Skill Reputation* model.

At LinkedIn, *Skill Reputation* model is used to recommend member with a certain skill expertise to recruiters. The accuracy of this model will be critical for the job recruiting process, which is the core value of LinkedIn. With the newly collected signals, we will focus the future work on developing a more robust and accurate *Skill Reputation* V2.0. To improve the quality of the collected signal, we may explore active learning when collecting the user response, i.e., by sampling the $\langle member, skill \rangle$ pairs that are less confident by our model, and injecting them to obtain user validation. Because we have multiple sources of signals, we can also use multi-task learning to combine different signals in a unified way. Each signal will be treated as a source of label in the multi-task learning model, and it will be straightforward to add new signals developed and collected by LinkedIn.

ACKNOWLEDGMENTS

The author would like to thank Cagri Ozcaglar, Myunghwan Kim, Wei Lu, Sen Yang for their generous comments on the paper, Rui Wang and Ying Han for helping with modeling, and Farzad H. Eskafi for coordinating the project.

REFERENCES

[1] [n. d.]. LinkedIn Official Website. https://news.stg.linkedin.com/about-us#statistics.

[2] Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. 2012. Discovering Value from Community Activity on Focused Question Answering Sites: A Case Study of Stack Overflow. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '12)*. ACM, New York, NY, USA, 850–858. https://doi.org/10.1145/2339530.2339665

[3] Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. 2012. Effects of User Similarity in Social Media. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining (WSDM '12)*. ACM, New York, NY, USA, 703–712. https://doi.org/10.1145/2124295.2124378

[4] Krisztian Balog and Maarten de Rijke. 2007. Determining Expert Profiles (With an Application to Expert Finding). In *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*. 2657–2662. http://ijcai.org/Proceedings/07/Papers/427.pdf

[5] Moira Burke and Robert Kraut. 2008. Mopping Up: Modeling Wikipedia Promotion Decisions. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work (CSCW '08)*. ACM, New York, NY, USA, 27–36. https://doi.org/10.1145/1460563.1460571

[6] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to Rank: From Pairwise Approach to Listwise Approach. In *Proceedings of the 24th International Conference on Machine Learning (ICML '07)*. ACM, New York, NY, USA, 129–136. https://doi.org/10.1145/1273496.1273513

[7] Olivier Chapelle and Ya Zhang. 2009. A Dynamic Bayesian Network Click Model for Web Search Ranking. In *Proceedings of the 18th International Conference on World Wide Web (WWW '09)*. ACM, New York, NY, USA, 1–10. https://doi.org/10.1145/1526709.1526711

[8] Albert C. Chen and Xin Fu. 2017. Data + Intuition: A Hybrid Approach to Developing Product North Star Metrics. In *Proceedings of the 26th International Conference on World Wide Web Companion (WWW '17 Companion)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 617–625. https://doi.org/10.1145/3041021.3054199

[9] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. ACM, New York, NY, USA, 785–794. https://doi.org/10.1145/2939672.2939785

[10] Cristian Danescu-Niculescu-Mizil, Gueorgi Kossinets, Jon Kleinberg, and Lillian Lee. 2009. How Opinions Are Received by Online Communities: A Case Study on Amazon.Com Helpfulness Votes. In *Proceedings of the 18th International Conference on World Wide Web (WWW '09)*. ACM, New York, NY, USA, 141–150. https://doi.org/10.1145/1526709.1526729

[11] Georges E. Dupret and Benjamin Piwowarski. 2008. A User Browsing Model to Predict Search Engine Click Data from Past Observations.. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '08)*. ACM, New York, NY, USA, 331–338. https://doi.org/10.1145/1390334.1390392

[12] Fredric C. Gey. 1994. Inferring Probability of Relevance Using the Method of Logistic Regression. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '94)*. Springer-Verlag New York, Inc., New York, NY, USA, 222–231. http://dl.acm.org/citation.cfm?id=188490.188560

[13] Carlos A. Gomez-Uribe and Neil Hunt. 2015. The Netflix Recommender System: Algorithms, Business Value, and Innovation. *ACM Trans. Manage. Inf. Syst.* 6, 4, Article 13 (Dec. 2015), 19 pages. https://doi.org/10.1145/2843948

[14] Laura A. Granka, Thorsten Joachims, and Geri Gay. 2004. Eye-tracking Analysis of User Behavior in WWW Search. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04)*. ACM, New York, NY, USA, 478–479. https://doi.org/10.1145/1008992.1009079

[15] V. Ha-Thuc, G. Venkataraman, M. Rodriguez, S. Sinha, S. Sundaram, and L. Guo. 2015. Personalized expertise search at LinkedIn. In *2015 IEEE International Conference on Big Data (Big Data)*. 1238–1247. https://doi.org/10.1109/BigData.2015.7363878

[16] Will Hill, Larry Stead, Mark Rosenstein, and George Furnas. 1995. Recommending and Evaluating Choices in a Virtual Community of Use. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '95)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 194–201. https://doi.org/10.1145/223904.223929

[17] Cho-Jui Hsieh, Mitul Tiwari, Deepak Agarwal, Xinyi (Lisa) Huang, and Sam Shah. 2013. Organizational Overlap on Social Networks and Its Applications. In *Proceedings of the 22Nd International Conference on World Wide Web (WWW '13)*. ACM, New York, NY, USA, 571–582. https://doi.org/10.1145/2488388.2488439

[18] Pei Lee, Laks V.S. Lakshmanan, Mitul Tiwari, and Sam Shah. 2014. Modeling Impression Discounting in Large-scale Recommender Systems. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)*. ACM, New York, NY, USA, 1837–1846. https://doi.org/10.1145/2623330.2623356

[19] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon.Com Recommendations: Item-to-Item Collaborative Filtering. *IEEE Internet Computing* 7, 1 (Jan. 2003), 76–80. https://doi.org/10.1109/MIC.2003.1167344

[20] Bo Pang and Lillian Lee. 2008. Opinion Mining and Sentiment Analysis. *Found. Trends Inf. Retr.* 2, 1-2 (Jan. 2008), 1–135. https://doi.org/10.1561/1500000011

[21] Tao Qin, Xu-Dong Zhang, De-Sheng Wang, Tie-Yan Liu, Wei Lai, and Hang Li. 2007. Ranking with Multiple Hyperplanes. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*. ACM, New York, NY, USA, 279–286. https://doi.org/10.1145/1277741.1277791

[22] Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. 2017. Snorkel: Rapid Training Data Creation with Weak Supervision. *Proc. VLDB Endow.* 11, 3 (Nov. 2017), 269–282. https://doi.org/10.14778/3157794.3157797

[23] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2000. Analysis of Recommendation Algorithms for e-Commerce. In *Proceedings of the 2Nd ACM Conference on Electronic Commerce (EC '00)*. ACM, New York, NY, USA, 158–167. https://doi.org/10.1145/352871.352887

[24] Ming-Feng Tsai, Tie-Yan Liu, Tao Qin, Hsin-Hsi Chen, and Wei-Ying Ma. 2007. FRank: A Ranking Method with Fidelity Loss. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*. ACM, New York, NY, USA, 383–390. https://doi.org/10.1145/1277741.1277808

[25] Yisong Yue, Thomas Finley, Filip Radlinski, and Thorsten Joachims. 2007. A Support Vector Method for Optimizing Average Precision. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*. ACM, New York, NY, USA, 271–278. https://doi.org/10.1145/1277741.1277790

[26] Yuchen Zhang, Weizhu Chen, Dong Wang, and Qiang Yang. 2011. User-click Modeling for Understanding and Predicting Search-behavior. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '11)*. ACM, New York, NY, USA, 1388–1396. https://doi.org/10.1145/2020408.2020613

[27] Guangyou Zhou, Siwei Lai, Kang Liu, and Jun Zhao. 2012. Topic-sensitive Probabilistic Model for Expert Finding in Question Answer Communities. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*. ACM, New York, NY, USA, 1662–1666. https://doi.org/10.1145/2396761.2398493