Salience and Market-aware Skill Extraction for Job Targeting

Baoxu Shi
LinkedIn Corporation, USA
dashi@linkedin.com

Jaewon Yang
LinkedIn Corporation, USA
jeyang@linkedin.com

Feng Guo
LinkedIn Corporation, USA
feguo@linkedin.com

Qi He
LinkedIn Corporation, USA
qhe@linkedin.com

Abstract

At LinkedIn, we want to create economic opportunity for everyone in the global workforce. To achieve this vision, it is crucial for LinkedIn to match job postings to quality applicants who are both qualified and willing to apply for the job. To serve this goal, LinkedIn offers two job targeting systems, namely Jobs You May Be Interested In (JYMBII) [13], which proactively target jobs to the quality applicants, and Job Search [19], which reactively recommend jobs that the job seeker qualifies.

Matching jobs to the quality applicants is a challenging task. Due to high cardinality (for LinkedIn, 645M members and more than 20M open jobs), it is computationally intractable to define job’s target member set by specifying individual members. For this reason, many models match candidates by profile attributes [1, 25, 37]. At LinkedIn, we mostly use titles and skills to target candidates (members). In other words, when a member comes to the job recommendation page, we recommend job postings whose targeting skills (or titles) match the member’s skills (or titles).

In this paper, we study how to target job postings by identifying relevant skills. Given a job posting, we extract skills from the job posting so that it can be shown to the members who have such skills. This task of mapping the job to a set of skills is very important because it determines the quality of applications for the job posting and affects hiring efficiency.

To improve the quality of applicants, what should be the objectives of extracting skills for job targeting? We argue that the extracted skills should not only be mentioned in the job posting, but also be relevant to the core job function. In other words, the skills should be salient to the job posting. Second, the extracted skills should be able to reach out to enough number of members. In other words, there should be enough supply for the skills in the job market. In summary, we aim to build a machine learning model to extract skills from job postings in a salience and market-aware way.

However, developing such salience and market-aware skill extraction model is a very challenging task. Not only because modeling the salience and market dynamic is hard, but also because the...
lack of ground-truth data. Given a job posting, it is tricky to get gold-standard labels for the skills that are both salient and have strong supply. One may propose using crowdsourcing to annotate job postings, but there are two problems. First, crowdsourcing may not be the most cost-efficient way to collect a large amount of training examples. Second and more importantly, labeling salient and high-supply skills requires very solid domain knowledge about the job market and the job posting itself. As we will show later, the labels collected by crowdsourcing do not give us the most salient skills with the best job market supply.

Figure 1 gives an illustration of the above challenge. Given a job posting, it is relatively straightforward for human to label skills mentioned in the posting (rectangles), and an annotator in the crowdsourcing platform would be able to do it. However, if the annotator aims to label which skills are more salient and have better job market supply, the annotator needs to understand both the job market and the job description well. For example, for the sake of market supply, the annotator should choose “deep learning” instead of specific tools such as “Pytorch” or “Keras”, but this is impossible if the annotator does not understand deep learning related skills and the deep learning job market. On the other hand, skills like “Communication Skill” have large supply but they are less salient in the context of deep learning engineer jobs.

Previous works usually treat this skill extraction task as a named entity recognition (NER) [24] task and use the standard named entity recognizer to identify skill mentions [19, 36]. In the example of Fig. 1, existing methods would identify all rectangles and use them for job targeting with equal importance. This would lead to showing this deep learning job to members who have “Communication Skills”. Since these methods do not consider the job market supply and salience of entities, they will lead to sub-optimal job targeting performance, as we will show later.

**Present work: Data collection.** In this paper, we tackle the problem of building salience and market-aware skill extraction model for job targeting. To collect the ground-truth labels, we note that the ground-truth skills need to meet dual criteria: salience and market-awareness. Since it is hard to come up with one data collection method to match both criteria, our strategy is to develop one method to cover each criterion, and combine the data collected using two methods. First option is focused on salience. We ask hiring experts to pick which skills they would want to use for targeting of their job postings. Since this method is from the salience perspective, we also conducted analysis to validate that the collected labels match our notions of market-aware skill extraction. This option allows us to collect explicit feedback from hiring experts, but this option is only available to a small portion of job postings that are created through the LinkedIn’s job creation flow.

To cover market-awareness and increase the training data size, we apply “distant supervision” to get weak labels from other job postings. In particular, for a given job posting, we track which members applied for the job and received positive feedback. We then identify common skills that these successful applicants have and use these skills as the ground-truth skills. We then perform analysis to validate that these skills are also salient.

**Present work: Modeling job market and entity salience.** After collecting the ground-truth, we develop Job2Skills, a novel machine learning model to extract skills in a market-aware, salience-aware way. We note that retraining existing NER models leads to sub-optimal results as they rely purely on the text information, ignoring important signals such as how much supply the skill has in the job market, how salient the skill is overall, and so on. Therefore, we add signals representing the job market supplies and the salience of skills in the model and significantly improve the performance in the offline evaluation and online A/B test.

**Present work: Product improvements.** We deploy the Job2Skills in production and improve the product metrics across multiple applications. First, we ask feedback from hiring experts in the job creation systems and outperform the existing NER-based model by more than 30%. Second, we employ this new skill extraction model in the job recommendation systems, one of the most important recommender systems at LinkedIn. We observe 2% member-job interaction and 6% coverage improvement in proactive job recommender (JMBII) system.

**Present work: Insights for Job Market.** We note that Job2Skills’ outputs capture the hiring trend in the job market for millions of companies that post jobs at LinkedIn. Since Job2Skills is trained on the feedback from hiring experts and job markets, it learns what kinds of talent that each employer is trying to recruit. We argue that Job2Skills reveals employers’ intention better than traditional information extraction methods that do not consider salience and market factors. We present extensive studies to showcase the power of insights that can be gained with Job2Skills. For example, we show that Job2Skills’ results can forecast that Macy’s would expand a tech office in a new location in two months before the official news article comes out. We also show that Job2Skills can vividly capture the rising and fall of popularity of a certain skill. Lastly we demonstrate that Job2Skills can be used to compare the required skill sets across different regional markets.

The contributions of this paper are summarized as follow:

- We propose a skill extraction framework to target job postings by skill salience and market-awareness, which is different from traditional entity recognition based method.
- We devise a data collection strategy that combines supervision from experts and distant supervision based on massive job market interaction history.
- We develop Job2Skills by combining NER methods with additional market and deep learning powered salience signals.
- We deploy Job2Skills to LinkedIn to improve overall hiring efficiency across multiple products.
- We provide a case study to show how the proposed market-aware skill extraction model yields better skill-related insights about the workforce and beyond.

2 PROBLEM DEFINITION

Here we will provide the formal definition of the job targeting task and show how we utilize the job targeting process to formulate our salience and market-aware skill extraction task.

**Definition 1. Job targeting** is an optimization task where given job posting $p$ and member set $M$, find a $t_m$-sized target candidate set $M_p \subseteq M$ of $p$ that can maximize the probability that member $m \in M_p$ belongs to the job posting $p$’s quality applicants set $A_p$, a
We can write the objective of Def. 1 as
\[
\arg\max_{S_p \subseteq S, |S_p| = t_s} \sum_{s \in S_p} I(s_m \cap S_p \neq \emptyset)Pr(m \in A_p),
\]
where \(I(\cdot)\) is an indicator function and \(S\) is the full skill entity set used by LinkedIn. Although Eq. 2 reduces the dimensionality of search space from 645 million members to 40 thousand skills, it is still combinatorial optimization and thus very hard to optimize directly. To make it more tractable, we make the following assumptions. We assume that each skill \(s\) has a utility \(U(m, p, s)\) that is the chance of \(m\) being qualified for \(p\) when \(m\) knows skill \(s\), and that the probability of being qualified \(Pr(m \in A_p)\) is the sum of utility \(U(m, p, s)\) of each skill \(s \in S_m \cap S_p\). In other words, we assume that each skill \(s\) increases the chance of \(m\) being qualified by \(U(m, p, s)\).
\[
Pr(m \in A_p) = \sum_{s \in S_p} I(s \in S_m)U(m, p, s).
\]
With this notation, Eq. 2 becomes
\[
\arg\max_{S_p \subseteq S, |S_p| = t_s} \sum_{s \in S_p} \sum_{m \in M_s} I(s \in S_m)U(m, p, s) = \sum_{s \in S_p} \sum_{m \in M_s} U(m, p, s).
\]
where \(M_s\) is members who have \(s\) in their skill set. We can further simplify the notation by introducing \(U(p, s) = \sum_{m \in M_s} U(m, p, s)\):
\[
\arg\max_{S_p \subseteq S, |S_p| = t_s} \sum_{s \in S_p} U(p, s),
\]
where \(U(p, s)\) is the sum of \(U(m, p, s)\) for all members who have \(s\). Given that \(U(m, p, s)\) is the increase in probability of \(m\) being qualified for \(p\) by knowing \(s\), \(U(p, s)\) quantifies the overall increase in qualified applicants by targeting members having \(s\). We call \(U(p, s)\) the skill \(s\)’s utility for job posting \(p\).

The formulation in Eq. 5 makes optimization much more tractable and simpler than the original form in Eq. 1. In Eq. 5, we optimize the sum of utilities \(U(p, s)\) for each skill \(s\). Therefore, choosing the optimal \(S_p\) can be done by picking \(t_s\) skills that have the highest value of \(U(p, s)\) for a given job posting \(p\). We call this problem salience and market-aware skill extraction. We name the problem in this way because we find that the skill \(s\) needs to satisfy the following two criteria to have high utility \(U(p, s)\).

1. **Skills should have sufficient market supply.** In order for \(U(p, s)\) to be high, the size of \(M_s\) needs to be large enough. In other words, there must be a sufficient number of members that have skill \(s\).

2. **Skills should be salient to the job posting.** Another factor that determines \(U(p, s)\) is the value of \(U(m, p)\). Remember that \(U(m, p, s)\) quantifies the chance that the member \(m\) is qualified for job posting \(p\) if the member has skill \(s\). If \(s\) is a core, “salient” skill for the job posting \(p\), it will have high \(U(m, p, s)\).

Combining the above two criteria together, we define the salience and market-aware skill extraction task as:

**Definition 2. Salience and Market-aware skill extraction** is an optimization task where given a job posting \(p\) and skill set \(S\), we estimate the utility \(U(s, p)\) that is the increase in qualified applicants by targeting members with \(s\).

As a comparison, we discuss the *salience- and market-agnostic approach* that chooses \(S_p\) by identifying the skills mentioned in the job posting content. We can think of it as solving Eq. 5 with modified utility \(U'(p, s)\) that purely depends on whether the skill is mentioned in the job posting. We can formally define it as follows:

**Definition 3. Salience- and market-agnostic skill extraction** is an optimization task where given a job posting \(p\) and skill set \(S\), extract \(t_s\)-sized skill set \(S_p \subseteq S\) for job posting \(p\) that maximizes the utility \(U'(s, p; \theta) \propto Pr(s|c_p)\), which is the likelihood that skill \(s\) is mentioned in \(p\)’s content \(c_p\).

As shown in Def. 3, this method simplistically defines a skill of a job posting by calculating the probability that it is mentioned in the job posting, using some named skill-entity recognizer. Because Def. 3 does not consider the skill salience and the market supply, the extracted skills are not for targeting quality applicants. Therefore Def. 3 is not solving the job targeting task defined in Def. 1.

Next, we will discuss the methods we use to gather the ground truth \(S_p\) for training our salience and market-aware skill extraction system, followed by how we learn the utility function described in Def. 2 for the proposed Job2Skills model.

### 3 DATA COLLECTION

As we state before, one of the major challenges of developing a salience and market-aware job targeting system is the lack of ground-truth data. Here we will describes two data collection approaches that address both salience and market-awareness.

**Collect from Job Posting Experts:** Although crowd-sourcing is scalable compared to labeling by group of experts, the quality is sub-optimal because non-experts cannot distinguish the level of skill...
salient and often ignore the market supply. Instead of designing a sophisticated skill labeling task for non-experts to collect high-quality data, we directly collect data from experts who select job targeting skills regarding the salience and job market implicitly.

To do so, we ask job posters to provide job targeting skills. Given a job posting \( p \), we collect the job poster saved job targeting skills to form the positive skill set \( S^+_p \), and use the rejected, recommended job targeting skills to create the negative set \( S^-_p \). Lastly, we construct the positive and negative job-skill training pairs \( J^+ = \{(p, s_i) | p \in P, s_i \in S^+_p \} \) and \( J^- = \{(p, s_i) | p \in P, s_i \in S^-_p \} \), respectively. We will refer this dataset as the Job Targeting skill (JT) dataset.

Fig. 2 shows the distribution of crowd-labeled (non-expert) and expert-labeled skills in terms of the number of LinkedIn members having the skill. We observe that expert labeled skills better align with the market supply in terms of member-side skill popularity.

Collect from Market Signals: Although hiring experts provide high quality data and address both entity salience and market-awareness, such approach is only available to a small portion of job postings that are created through the LinkedIn’s job creation flow. To cover more jobs and generate a large amount of labeled data for model training, we decide to get a large amount of weak labels using job market signals instead of human annotations.

To be specific, given job posting \( p \), we first collect \( A'_p \) by assuming quality applicants are the members who apply for job \( p \) and receive positive interactions from the recruiters. Note that \( A'_p \) is an approximation of the true quality applicant set because we only consider one stage of the recruiting process on LinkedIn. With \( A'_p \), we define \( S^*_p = S^-_p \cap A'_p \), where \( S^*_p \) is a set of common skills shared by \( A'_p \) and \( S^-_p \) are the skills mentioned in \( p \). Similarly, \( S^-_p = S^*_p \setminus A'_p \), where the negative skills are the ones mentioned in the job posting but not shared by the quality applicants \( A'_p \). We will refer this dataset as the Quality Applicant skill (QA) dataset.

To examine the salient level of these market signal derived skills, we sample jobs with labeled skills from both quality applicants and job targeting datasets, and then compare the top skills shared by quality applicants to the skills labeled by the job posters. We find that the top-ranked quality applicant skills largely overlap with the job poster selections. To be specific, for the top-5 quality applicant shared skills, more than 60% of them are labeled as positive skills and 8% of them are labeled as negative skills by job posters.

4 THE PROPOSED JOB2SKILLS MODEL

After we describe the procedure we use to collect the ground truth for salience and market-aware job targeting skill extraction, here we discuss how we build the proposed Job2Skills using multi-resolution skill salience features and market-aware signals. Compared to simple skill tagging, which merely identifies mentioned skills, multi-resolution skill salience will identify important skills from all mentioned skills.

4.1 Multi-resolution Skill Salience Feature

In this work, we hypothesize that good job targeting skills should be salient to the job posting. Unlike other text, job postings are usually long text with several well-structured segments, e.g., requirements, company summary, benefits, etc. To accurately estimate the level of skill salience and fully utilize the rich job posting information, one should not only consider the mentioning of the skill, but also other segments in the job and the entire job posting.

Next, we will first briefly describe how we tag skills from job postings, and then provide details on how we explicitly model the skill salience at three resolution levels: sentence, segment, and job level. Fig. 3 gives an overview of the multi-resolution skill salience features we used in this work.

To identify skills from job postings, we first utilize an existing, in-house skill tagger to find out all possible skill mentions. By leveraging a comprehensive skill taxonomy with an extensive set of skill surface forms, the skill tagger can identify the majority of the mentioned skills. We then pass the skill mentions into a feature-based regression model to link them to the corresponding skill entities. After we find all the skills in the job posting, we then model the skill salience from the following three levels:

Sentence-level Salience: To estimate the skill salience at sentence level, we build a neural network model to learn the skill salience by modeling the job posting sentence that contains the skill mention and the skill’s surface form. The model is defined as

\[
Pr(s|c_{sent}) = \sigma(W \cdot f_{encoder}(c_{sent}, s_{sent}) + b),
\]

where \( \sigma(x) = 1/(1 + \exp(-x)) \), \( s \) is some skill, \( c_s \) is the skill’s surface form text, \( c_{sent} \) is the sentence containing the skill mention, and \( f_{encoder} \) is some text to embedding encoder. We tested multiple encoders including FastText [5], Universal Sentence Encoder (USE) [7], and BERT [12]. For FastText and USE, we encode the skill and sentence separately and then use the concatenated embedding as the encoder output. For BERT, we feed both skill and sentence into the model and pick the embedding of the [CLS] token as the encoder output.

Segment-level Salience: Unlike sentence level skill salience, where the input text length is limited to a single sentence, job segment (one or multiple consecutive paragraphs describing the same topic), e.g., the company summary segment or requirements segment in a job description, are usually longer and much noisier. Therefore, it is not easy to model them directly using the neural network models that are designed for shorter text. Instead of modeling the job
segment text directly, we choose to represent a job segment by the embeddings of the skill entities mentioned in the segment.

To get the entity (title and skill) embeddings \([31, 34]\), we learn LinkedIn’s unsupervised entity embeddings using LinkedIn member profile. The skip-gram loss is defined as:

\[
\mathcal{L} = \sum_{m \in M} \sum_{e \in E_m} k_{pos} \sum_{c \in C} \mathbb{E}_{e \sim \text{Unif}(E_m) \setminus \{e\}} \left( \log(\sigma(W_{e} \cdot W^{T}_e)) \right)
\]

\[+ \sum_{c \in C} \mathbb{E}_{e \sim \text{Unif}(E_m)} \log(\sigma(-W_{e} \cdot W^{T}_e)), \]  

(7)

Where \(k_{pos}\) and \(k_{neg}\) are the number of positive and negative samples, and Unif is a uniform sampling function. Simply put, for each LinkedIn member \(m\)’s profile entity \(e \in E_m\), we optimize the entity embedding \(W_e\) so that the skills and titles appear in the same member’s profile entity set \(E_m\) are similar to each other comparing to a random entity \(e \in E\) that is not in \(E_m\).

After we obtain the entity embedding matrix \(W\), we define the segment-level skill salience as:

\[
Pr(s|E_p) = \sigma(f_{\text{meanpool}}(\{W_{s}, |j \in S_{c,p} | \}) \cdot W^{T}_{T}),
\]

(8)

where \(S_{c,p}\) is set of skills mentioned in the job content \(c_{p}\) with segment label \(I\). This measures the similarity between the given skill \(s\) and the centroid of other skills mentioned in segment \(I\).

**Job-level Salience:** Similar to segment level salience, Job level skill salience is modeled by the average embedding similarity between the given skill and all entities mentioned in the job posting, which includes skill and title entities. The salience score is defined as:

\[
Pr(s|E_p) = \frac{1}{|E_p|} \sum_{e \in E_p} \sigma(W_{s} \cdot W^{T}_e)
\]

(9)

where \(E_p\) is a set of title and skill entities mentioned in job \(p\).

Note that we choose to compute entity-wise similarity instead of meanpooled-similarity here because 1) the size of \(E_p\) is often significantly larger than the size of \(S_{c,p}\) and may contain more noisy data, and 2) \(E_p\) contains different types of entities from multiple aspects of the job posting. Based on above observation, we believe it is sub-optimal to compute job-level salience by forming a single meanpooled centroid using all mentioned entities.

In sum, for a skill candidate \(s\) of a job posting, we use the above methods to compute multi-resolution salience probability scores \((Pr(s|c_{sent}), Pr(s|S_{c,p}), \text{and } Pr(s|E_p))\) as salience features, combine them with the market-aware signals which we will cover in the next section, and then build the final salience and market-aware skill extraction model.

### 4.2 Market-aware Signals

Besides the salient level of entities, we also hypothesize that good job targeting skills should have sufficient market supply. To model the supply of skills, it is necessary to factor in market-related signals into the proposed **Job2Skills**. In general, the market-related signals can be derived from LinkedIn’s member base \(M\) and job postings \(P\).

Next, we will describe the signals from these two groups.

**Member Features:** The goal of the member feature group is to capture how skills can reach a broader audience by measuring member-side skill supply, and therefore to improve job exposure.

Here we consider both the general skill supply which measures the overall skill popularity among all members \(Pr(s|M)\), and the cohort affiliations, which indicate the skill supply with finer granularity. To be specific, we partition the member set \(M\) into a group of non-empty member subsets (cohorts) \(M\) using different strategies, then compute the point-wise mutual information (PMI) and the entropy \((H)\) of \(s\) given a cohort \(M_i\) as follows:

\[
\text{PMI}(M_i; s) = \log \frac{Pr(s|M_i)}{Pr(s|M)} \cdot H(s) = - \sum_{M_j \in M} \frac{Pr(s|M_j)}{Pr(s|M)} \log \left( \frac{Pr(s|M_j)}{Pr(s|M)} \right),
\]

(10)

in which \(\bigcup_{M_i \in M} M_i = M\). The partition \(M\) can be created using one or multiple member attributes. By combining multiple attributes, the model can detect subtle skill supply differences. For example, by grouping members using both industry and job title, we can discover that although skill KDB+, a financial database, is not popular among either software developers or in the financial industry, it is a preferred skill in cohort software developers & financial industry.

**Job Features:** Because demand implicitly influences the supply, here we also measure the skills’ demand in terms of the job-side skill popularity \(Pr(s|P)\). Similar to member features, here we use pointwise mutual information \(\text{PMI}(I; s)\) to model the job-side skill popularity, where \(I\) is the job posting segment label, e.g., summary and requirement.

### 4.3 Job2Skills

After discussing both salience and market-aware features, now we will discuss how we train the proposed **Job2Skills** model using the generated features. Recall that we need to learn the utility function defined in Def. 2 to infer salience and market-aware skills for job targeting. With the job-skill pairs \(J^+\) and \(J^-\) that we collected from job posters and features described in Sec. 4, we can learn the utility function \(U\) by viewing this task as a binary classification problem where for a given skill \(s\) and a job posting \(p\), predict if \(s\) is a salience and market-aware job targeting skill for \(p\).

Among all the possible machine learning models ranging from the generalized linear model to neural networks \([9, 15, 22, 44]\), here we chose to use XGBoost \([8]\) because it is fast, interpretable, and has good scalability with small memory footprint. By leveraging an in-house implementation of XGBoost, we were able to serve the model online without noticeable latency increase over the existing linear production model. The XGBoost-based **Job2Skills** is trained with a logistic regression loss to optimize the binary classification task, and we use the resulting tree-based **Job2Skills** model as the utility function \(U\) to extract market-aware job targeting skills for job postings. We define the loss function of the **Job2Skills** as:

\[
L = - \sum_{(p, s) \in J^+} \log \left( \sum_k f_k \left( \phi \left( s, c_{p}, M, P, S_{c,p} \right) \right) \right) \]

\[- \sum_{(p, s) \in J^-} \log \left( 1 - \sum_k f_k \left( \phi \left( s, c_{p}, M, P, S_{c,p} \right) \right) \right) + \sum_k \Omega(f_k) \]

(11)

in which \(J^+\) and \(J^-\) are the positive and negative job-skill pairs, \(\phi(\cdot)\) denotes the combined market and salience feature vector of a given \((p, s)\) pair. \(f_k\) represents the \(k\)th tree in the model, and \(\Omega(f_k) = \gamma T + \frac{1}{2} \lambda \|w\|^2\) is the regularization term that penalizes the
complexity of tree $f_k$, in which $T$ denotes the number of leaves in tree $f_k$, $w$ is the leaf weights, $\gamma$ and $\lambda$ are the regularization parameters.

5 EXPERIMENTS

In this section, we conduct an extensive set of experiments with both offline and online A/B tests to demonstrate the effectiveness of the proposed Job2Skills model compared to our market-agnostic production model. We also present a case study to demonstrate the actual skills returned by Job2Skills and how we can get better market insights from it.

The Job2Skills model evaluated in this section contain all aforementioned market-aware and multi-resolution skill salience features. Note that the job-level salience sub-model we used in production Job2Skills model is a FastText-based model instead of the BERT model we tried offline. This is because we observed significant latency reduction with only 3% salience accuracy drop.

The market-agnostic production model (baseline for short) we compared against is a logistic regression model trained with skill appearance features, e.g. job-level features such as is the skill mentioned in the text?, where the skill is mentioned?, and global-level features such as mention frequency.

The offline training and evaluation data are collected using the following procedure. We used 16 months of LinkedIn’s English Premium jobs posted on LinkedIn as input and generated around 3 million job-skill pairs for training and evaluation. To be specific, we used the methods described in Sec. 3 and collected 1) job-targeting (JT) dataset using job poster provided job targeting skills, and 2) quality applicant (QA) dataset using the common skills shared by job applicants who received positive feedbacks from recruiters. We used 60% of them for training, 20% for validation, and the rest for testing. Note that unlike Job2Skills, the production baseline is trained on the JT data only. During inference, both methods use the same skill tagger to get the same set of skill candidates from jobs.

5.1 Offline Evaluation

We present the Job2Skills offline evaluation result on the hold-out sets of the two training datasets (JT and QA), and report the relative AUROC improvement against the production baseline model. As shown in Tab. 1, Job2Skills significantly outperforms baseline by 55.77% on the job targeting (JT) set and 49.27% on the quality applicants’ (QA) skill set. Moreover, by training with both human-labeled JT and derived QA dataset, we are able to generalize the model and achieve a better overall AUROC on both tasks by increasing the overall AUROC by +62.67%.

Next, we present an ablation study to learn the importance of each feature group and present the relative AUROC improvement in Tab. 2. Note the evaluation dataset used here is a slightly different JT dataset collected using the same procedure as Tab. 1 but different time span. We can see both salience and market feature group positively contribute to the model performance improvement. We also observe that when only using one feature group, model trained with deep learning-based salience feature is 0.95% better than market-feature only model. By combining both group of features, we further improve the AUROC by 1.56% comparing to the salience only model. These improvements indicate that market-dynamics modeled by market-aware features provide additional information on the skill importance for job targeting that cannot be captured by modeling job posting-based skill salience only.

In addition to the ablation study, we also looked at the feature importance of the Job2Skills model trained using both market and salience features. We found that all three salience features are ranked within the top-5% most important features, and segment-level salience feature is the most important one followed by the sentence-level and job-level salience features. This means both deep learning powered salience features and market features are crucial to the model and cannot be replaced by each other.

5.2 Online Job Recommendations

In this section, we deploy Job2Skills to production, apply it to all LinkedIn Jobs to extract skill entities for job targeting, and retrain our job recommender system. Jobs You May Be Interested In (JYMBII [18]), based on the extracted salience and market-aware job targeting skills. We perform online A/B test with 20% of the LinkedIn traffic for 7 days, and observe significant lift in multiple metrics. As shown in Tab. 3, the Job2Skills-based JYMBII model not only recommends better jobs (reflected by increased job apply and save rate), but also increases the percentage of members receive job recommendations.

| Model | AUROC Improvement (%) |
|-------|------------------------|
| Job2Skills w/ Market and Salience features | 56.99% |
| – Salience features only | 55.43% |
| – Market features (member+job) only | 54.91% |
| – Member features only | 46.47% |
| – Job features only | 49.88% |

| Model | Onsite Apply | Job Save | Member Coverage |
|-------|-------------|----------|-----------------|
| Job2Skills | +1.92% | +2.66% | +6.71% |
### 5.3 Online Job Targeting Skill Suggestions

We also apply Job2Skills to provide job targeting skill suggestions in LinkedIn’s job posting flow. When a recruiter posts a job on LinkedIn, we use Job2Skills to recommend 10 skills, and the poster will be able to save at most 10 job targeting skills by either selecting from the recommendation or providing their own. We ramped our Job2Skills model to 50% of LinkedIn’s traffic, and report the 4-week A/B test result. The definition of the metrics are as follows:

- **Skill add rate**: % of skills that are not recommended and are manually added.
- **Skill reject rate**: % of recommended skills that are rejected.

As shown in Tab. 4, the skills recommended by Job2Skills are notably better than the existing production model because the recruiters are now 31.44% less likely to manually add a job targeting skill and 33.71% less likely to reject recommendation. In general, the Job2Skills increases the overall job targeting skill coverage and quality by adjusting skill importance as a function of the skills’ salience and market signals.

| Model                        | Skill Add Rate | Skill Reject Rate |
|------------------------------|----------------|------------------|
| Job2Skills (market-aware)    | −31.44%        | −33.71%          |
| Job2Skills (market- and salience-aware) | −33.75%       | −37.06%          |

### 6 QUALITATIVE ANALYSIS

We have shown Job2Skills improves job targeting at LinkedIn because it captures salience and market supplies. Here we further demonstrate salience and market-aware skills’ strength in market analysis. We claim that Job2Skills can capture hiring trend — e.g., what skill sets are required in different sectors? what kinds of talents employers are recruiting? — very well. As we explained, Job2Skills is trained on the hiring experts feedback and market supplies. In other words, it is trained on signals representing what kinds of people (skill sets) the companies want the most, and what kinds of people that actually got hired. Next we show that Job2Skills’ results can reveal the intent of employers better than the baseline model which based on named entity recognition.

### 6.1 Skill Trend Insight

The first kind of cases is about the trend of skills. Which skills are getting popular? Which company is growing a certain job segment?

Figure 4: % of Macy’s jobs posted per month from Jun. to Dec. 2019 that require software development related skills.

We show that Job2Skills’ results can be used to forecast this trend even before actual news articles coming out.

In 2019, we identified Macy’s tech expansion using the results of Job2Skills. In Fig. 4, we presented the percentage of Macy’s monthly posted jobs that require skills such as Java, Javascript, and SQL. We found Macy’s demand on tech skills almost tripled from June to December, 2019. We suspect such radical change indicates Macy’s is planning to invest into technology sector. Two months after we detected this trend, in February, 2020, Macy’s officially announced its tech operation expansion in Atlanta and New York.

Besides predicting company expansions, Job2Skills also better captured the skill supply and demand regarding market dynamics such as recruiting circles. We present the skill popularity trend of Azure, the fifth trending skill in SF Bay Area, in Fig. 5. We found that job targeting skill popularity is highly correlated with major hiring and vacation seasons. Interestingly, Job2Skills popularity changes also correlated with other market signals such as company performance. The first gray box in Fig. 5 highlights a significant four-month increase trending from Dec. 2017 to Mar. 2018 of Azure, which aligns with MSFT’s 2018Q3 earning that shows Azure cloud has a 93% revenue growth. We suspect the correlation is caused by the adoption of Azure services in the market – more companies are using Azure hence the skill becomes more popular, and such market share increase also reflect in the revenue growth.

### 6.2 Skill Insights

The second kind of cases is revealing diversified skill demands in different industry sections and regions. We show that the job targeting skills generated by Job2Skills better captures such market diversity by modeling salience and market signals jointly.

In Tab. 5 we present the top-10 job targeting skills of three occupations, Software Engineer, Data Scientist, and Audit Tax Manager. It is clear that unlike the salience and market-agnostic model which mainly focuses on very specific skills with limited supplies such as C++0x and DropWizard, Job2Skills is able to return a diversified set of skills at the right granularity ranging from popular programming languages to soft skills such as communication and management.

Besides better representing skills for different occupations, the skills generated by Job2Skills can also capture skill supply and demand differences in different industries and regions. Here we compared top-5 job targeting skills between government and technology industries in the United States and India. As shown in Tab. 6, skills returned by the proposed Job2Skills is significantly different from the market and salience agnostic baseline model. We believe this

---

1. https://www.bloomberg.com/news/articles/2020-02-04/macy-s-to-move-san-francisco-tech-offices-to-new-york-atlanta
Table 5: Sample of top-10 job targeting skills extracted by the salience and market-agnostic baseline and Job2Skills.

| Job Title                  | Skills returned by Both Models | Skills returned by Job2Skills only |
|---------------------------|-------------------------------|-----------------------------------|
| Software Engineer         | Design, Java, Communication, C, C+++, Management, Javascript, SQL, Cloud Computing, Architecture | OpenGL, DropWizard, ActiveRecord, LLVM, Sinatra, C++, Guice, GRPC, cmake, Boost C++ |
| Data Scientist            | Data Science, Machine Learning | Data Mining, Python, Analytics, Pattern Recognition, Statistics, R, AI, Communication |
| Audit Tax Manager         | Auditing                      | Communication, Management, Research, Tax Preparation, Supervisory Skills, Presentations, Tax Compliance, Engagements, Budgeting |

Table 6: Sample of top-5 job targeting skills per industry and country extracted by the salience and market-agnostic baseline and Job2Skills. Red squared skills are sub-optimal.

| Industry  | United States | India |
|-----------|---------------|-------|
| Government |               |       |
| teaching  | analytical skills | start-ups, communication |
| management | communication | management, think tank |
| fire control | defense | procurement, research |
| law enforcement | hazardous materials | finance, social media |
| Technology |               |       |
| management | sales | management, project management |
| sales | cloud computing, communication | sales |
| Salesforce.com | problem solving | Salesforce.com, sales |
| Jakart EE | customer experience | solution architecture |
| cloud computing | sales | business process |
| consulting | Salesforce.com | helping clients |

is because Job2Skills considers skill salience and market dynamics. For example, the baseline model wrongly pick TSO (Time Sharing Option) as a top skill in US government industry because it is mentioned in many jobs posted by Transportation Security Administration and The Department of Homeland Security. However, the TSO mentioned in those jobs actually refers to Transportation Security Officer. By evaluating the skill salience, Job2Skills was able to identify TSO is not a valid skill. In addition, the market and salience agnostic baseline also ranked many specific tools such as SAP and Jakarta EE as top skills in Indian technology industry. Instead of targeting on very specific skills, the proposed Job2Skills measures the supply in the Indian market and selected many job-related, high supply skills such as customer experience and helping clients.

The skills extracted by Job2Skills can reveal market insights due to its market-awareness. It is interesting that the government industry market of US is significantly different from India. While India has a focus on a variety of skills ranging from research to social media to rights, US mostly focuses on military related skills. We suspect this is because US government positions are mostly defense/environmental related positions and prefer veterans. The technology industry is quite different between US and India, too. As shown in Tab. 6, it is clear that the Indian market has a focus on IT support whereas US is more about management and sales.

Moreover, we found that Job2Skills captures the skill differences between seniority levels. In Tab. 7, we presented the top-5 job targeting skills of US government and technology industries generated by the baseline and the proposed Job2Skills. Compared to the baseline, skills generated by Job2Skills are more representative and capture the skill shift across different seniority levels. For example, entry level positions require domain-specific skills such as SQL and Java, and higher level roles, regardless the industry, focus more in management skills such as management and leadership. The baseline model also selects many less relevant skills such as DES (Data Encryption Standard) and management for entry-level technology role due to the lack of salience and market modeling.

Table 7: Top-5 job targeting skills of US government and technology generated the salience and market-agnostic baseline and Job2Skills. Red squared skills are sub-optimal.

| Industry  | United States | India |
|-----------|---------------|-------|
| Government |               |       |
| DES        | management    | DES |
| project management | leadership | project management |
| communication | management | interpersonal skills |
| Technology |               |       |
| sales      | cloud computing | sales |
| management | consulting | cloud computing |
| Salesforce.com | sales | consulting |
| Jakarta EE | software development | sales |
| cloud computing | management | software development |

7 RELATED WORK

Job Recommendation. Previous work usually treat the job targeting problem as job recommendation [14, 37], and optimizes the model using direct user interaction signals such as click, dismiss, bookmark, etc. [2, 3, 16, 40]. Borisuyk et al. proposed Lijar [6] to redistribute job targeting audiences and improve marketplace efficiency. Dave et al. designed a representation-learning method to perform job and skill recommendations [11]. Li et al. used career history to predict next position [20]. None of the previous works address the most pressing job targeting issue, which is how to properly represent jobs with relevant, important attribute entities to improve the number of quality applicants a job can reach.

Skill Analysis. Traditionally, skill analysis are often conducted by experts manually to either gain insights [27] or curate structured taxonomy [10]. Recently, SPTM [41] used topic modeling to measure the popularity of 1,729 IT skills from 892, 454 jobs. TATF [39] is a trend-aware tensor factorization method that models time-aware skill popularity. DuerQuiz [29] is proposed to create in-depth skill assessment questions for applicant evaluation. These methods were applied to small-sacle IT jobs only and are not designed to extract skills for job targeting purpose. Recently Xiao proposed a social signal-based method for members’ skill validation [42]. However it is not applicable to jobs due to the lack of such signals.

Job Market Analysis. Modeling job targeting and recommendation using skills is mostly inspired by economic research which analyzes the labor market using skills as the most direct and vital signal [4, 32]. However these works are either conducted on a very small scale or using only a handful of hand-crafted general skill categories. Woon et al. [38] performed a case study to learn occupational skill changes, but the skills are limited to 35 skills provided by O’NET [26]. Radermacher et al. [30] studied the skill gap between fresh graduates and industry expectations based on the feedback of 23 managers and hiring personnel using 16 hand
picked skills. Recently, [17, 36] analyzed labor demand and skill fungibility using a skill taxonomy with engineered features. Lastly, we conducted extensive experiments and showed that Job2Skills significantly improves the quality of multiple LinkedIn products including job targeting skill suggestions and job recommendation. We also performed large-scale case studies to explore interesting insights we obtained by analyzing Job2Skills results. In future work, we plan to add temporal information into the model and explore advanced methods to learn skill embeddings.

8 CONCLUSION AND FUTURE WORK

In this work, we proposed salience and market-aware skill extraction task, discussed two data collection strategies, and presented Job2Skills, which models skill salience using deep learning methods and market supply signals using engineered features. Lastly, we conducted extensive experiments and showed that Job2Skills significantly improves the quality of multiple LinkedIn products including job targeting skill suggestions and job recommendation. We also performed large-scale case studies to explore interesting insights we obtained by analyzing Job2Skills results. In future work, we plan to add temporal information into the model and explore advanced methods to learn skill embeddings.

REFERENCES

[1] Fabian Abel. 2015. We know where you should work next year: job recommendations. In RecSys.
[2] Fabian Abel, Yashar Deldjob, Mehdi Elahi, and Daniel Kohlendorf. 2017. RecSys challenge 2017: Offline and online evaluation. In RecSys.
[3] Deepak Agarwal, Bee-Chung Chen, and Pradheep Elango. 2009. Spatio-temporal models for estimating click-through rate. In WWW.
[4] David H Autor, Frank Levy, and Richard J Murnane. 2003. The skill content of recent technological change: An empirical exploration. The Quarterly Journal of Economics 118, 4 (2003).
[5] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. TACL 5 (2017).
[6] Fedor Borisyuk, Liang Zhang, and Krishnaram Kenthapadi. 2017. LiJAR: A system for job application redistribution towards efficient career marketplace. In KDD.
[7] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St Chuan. 2017. Neural collaborative filtering. In WWW. The impact of Person-Organization Fit on Talent Management: A Structure-Aware Convolutional Neural Network Approach. In KDD.
[8] Shirish Vasudevan, Moninder Singh, Joydeep Mondal, Michael Peran, Benjamin Zweig, Brian Johnston, and Rachel Rosenfeld. 2018. Estimating Fungibility Between Skills by Combining Skill Similarities Obtained from Multiple Data Sources. Data Science and Engineering 3, 3 (2018).
[9] Rohan Ramanath, Hakan Inan, Gunog Polatkan, Bo Hu, Qi Guo, Cagri Ozcaglar, Xiannen Wu, Krishnaram Kenthapadi, and Sahin Cem Geyik. 2018. Towards Deep and Representation Learning for Talent Search at LinkedIn. In CIKM.
[10] National Research Council et al. 2010. A database for a changing economy: Review of the Occupational Information Network (O*NET). 2003.
[11] Vachik S Dave, Baichuan Zhang, Mohammad Al Hasan, Khalifeh AlJadda, and Mohammed Korayem. 2018. A combined representation learning approach for better job and skill recommendation. In CIKM.
[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018).
[13] Yupeng Gu, Bo Zhao, David Hardtke, and Yizhou Sun. 2016. Learning global term weights for content-based recommender systems. In WWW.
[14] Cheng Guo, Hongyu Lu, Shaojun Shi, Bin Hao, Bin Liu, Min Zhang, Yiqun Liu, and Shaoping Ma. 2017. How Integration helps on Cold-Start Recommendations. In RecSys Challenge 2017.
[15] Xiangnan He, Lin Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In WWW.
[16] Chao Huang, Xian Wu, Xuchao Zhang, Chuxu Zhang, Jiashu Zhao, Dawei Yin, and Nitesh V Chawla. 2019. Online Purchase Prediction via Multi-Scale Modeling of Behavior Dynamics. In KDD.
[17] Brian Johnston, Benjamin Zweig, Michael Peran, Charlie Wang, and Rachel Rosenfeld. 2017. Estimating skill fungibility and forecasting services labor demand. In Big Data.