Identifying Decision Makers from Professional Social Networks

Shipeng Yu
LinkedIn, Inc.
siyu@linkedin.com

Evangelia Christakopoulou
University of Minnesota
evangel@cs.umn.edu

Abhishek Gupta
LinkedIn, Inc.
agupta@linkedin.com

ABSTRACT

Sales professionals help organizations win clients for products and services. Generating new clients starts with identifying the right decision makers at the target organization. For the past decade, online professional networks have collected tremendous amount of data on people's identity, their network and behavior data of buyers and sellers building relationships with each other for a variety of use-cases. Sales professionals are increasingly relying on these networks to research, identify and reach out to potential prospects, but it is often hard to find the right people effectively and efficiently. In this paper we present LDMS, the LinkedIn Decision Maker Score, to quantify the ability of making a sales decision for each of the 400M+ LinkedIn members. It is the key data-driven technology underlying Sales Navigator, a proprietary LinkedIn product that is designed for sales professionals. We will specifically discuss the modeling challenges of LDMS, and present two graph-based approaches to tackle this problem by leveraging the professional network data at LinkedIn. Both approaches are able to leverage both the graph information and the contextual information on the vertices, deal with small amount of labels on the graph, and handle heterogeneous graphs among different types of vertices. We will show some offline evaluations of LDMS on historical data, and also discuss its online usage in multiple applications in live production systems as well as future use cases within the LinkedIn ecosystem.

Categories and Subject Descriptors

I.2 [**Artificial Intelligence**]: Miscellaneous

General Terms

Algorithms, Theory, Performance

Keywords

Decision Makers, Social Network Mining, Graph Mining

1. INTRODUCTION

As a crucial part of any for-profit organizations, sales professionals help to win customers for products and services. For B2B selling,[1] their job typically starts from identifying the right decision makers at the client organizations. This process is often called *prospecting* or *lead generation*. The ability to do this well is arguably the most important skill set that any successful sales professional need to have [6, 10, 12].

As we have entered into the social era, sales professionals started to pay more attention to developing relationships with potential clients as part of the sales process. This is often referred to as *social selling*, and includes components such as social prospecting, personal branding, employee advocacy, and social relationship management.[2] Among these components, *social prospecting* involves monitoring and searching social networks for signs of customer interest, immediate buying intent, or qualified prospect status based on industry, role, geography, etc. It's the entry point of identifying potential clients in the modern sales process.

Nowadays, social prospecting often takes place via social networks such as LinkedIn, Twitter and Facebook. Sales professionals are increasingly checking on member's profile, social connections, and blogs/updates on these social networks. However, they often find it hard to identify prospects they are interested in, despite the large amount of data available in various sources. They need intelligent data tools to help them find the target clients effectively and efficiently.

In this paper we present LDMS, the LinkedIn Decision Maker Score, to capture the fundamental notion of *decision makers* in the sales context by leveraging the professional members data on LinkedIn. The goal of LDMS is to consolidate all member's profile data with social graphs data and calculate a score for each of the 400M+ LinkedIn members, quantifying his or her ability of make or influence a sales decision. We will specifically discuss the modeling challenges of developing LDMS, and present two approaches to tackle the problem, a graph summarization approach and a bipartite graph learning approach. Both approaches are able to leverage both the graph information and the contextual information on the vertices, deal with small amount of labels on the graph, and handle heterogeneous graphs among different types of vertices. These methods are discussed in the LDMS context, but they can be applied to many other ranking or influence extraction problems on social networks or

[1] Business to Business Selling, `https://en.wikipedia.org/wiki/Business-to-business`

[2] `https://en.wikipedia.org/wiki/Social_selling`

graphs in general. We will show performance of both approaches on LinkedIn data, and also discuss how LDMS is currently used in LinkedIn production systems.

The rest of the paper is organized as follows. In Section 2 we cover the related work on graph-based learning methods for influential node discovery, and in Section 3 we present the overview of the problem and the general setup. Then in Section 4 we present two learning approaches to tackle this challenging problem, and in Section 5 we show some experimental results on some offline data. Finally we discuss applications of LDMS within the LinkedIn ecosystem in Section 6, and conclude the paper in Section 7.

2. RELATED WORK

There has been a rich literature on identifying important or influential nodes in a network, both in an unsupervised setting and a semi-supervised setting. We briefly describe some of the work in this section, and also discuss the importance of decision maker identification for the sales use case.

2.1 Unsupervised graph-based ranking

The identification of decision makers can be viewed as finding the most important nodes in a graph. If treated as an unlabeled problem, the most popular graph-based algorithms are PageRank [2] and HITS [11], both in the context of web page ranking. PageRank is an iterative algorithm behind Google search engine, in which the rank of a web page is the probability that a random surfer would visit that page starting from a random page. HITS, on the other hand, tries to answer the question of how authoritative a given web page is. The model assumes that there exist authority pages and pages that link to many related authorities, referred to as hubs, interconnected by a mutually reinforcing relationship.

Several algorithms have been proposed to enhance PageRank and HITS. Haveliwala et al. [8] presented different approaches to personalize PageRank. TrustRank [7] was proposed for spam detection using a seed of expert-verified trusted pages. BrowseRank [14] also incorporates the time component by taking into account how much time users spend on a web site. It uses continuous time Markov process instead of the discrete time Markov process that PageRank uses. An application of PageRank-based algorithm for identifying trendsetters was presented by Trumper et al. [18]. They identify the trendsetters in information networks by using temporal attributes of nodes and edges, and rank users based on their ability to promote new ideas which will be adopted by many other users later.

2.2 Context-aware graph algorithms

Instead of only considering the link structure of the network, several papers also investigated how to incorporate contextual attributes of the nodes into learning and ranking. Topic-sensitive PageRank [9] pre-computes a set of biased PageRank vectors using a set of representative topics. It then uses these to generate query-specific importance scores at query time. Richardson et al. [17] proposed an intelligent surfer which is a query-dependent, content-sensitive version of PageRank. The surfer probabilistically hops from page to page, depending on the content of the page and the query terms the user is searching. TwitterRank [19] ranks users by taking into account the topical similarity between users based on their tweets besides the graph information. The random surfer performs a topic-specific walk and then the overall influence is measured by aggregating the topic-specific TwitterRank over all the topics. Pal et al. [16] identified authorities in Twitter by performing probabilistic clustering over the feature space and then rank the users inside the cluster, under the assumption that features follow a Gaussian distribution.

2.3 Bipartite graph mining

Several work explored the bipartite graphical structure to identify important nodes. The majority of them have been developed in the context of co-authorship and citation networks. Zhou et al. [23] proposed an approach to combine two random walks, one as intra-class walk between same-type entities (authors or documents) and the other as inter-class walk across the two sides of the bipartite graph. The random surfer has a certain probability of taking intra- or inter-class steps at any vertex. Meng et al. [15] developed a rule-based method which iteratively ranks authors and publications following a mutual-reinforcement strategy until convergence. There has also been work on influence mining on a heterogeneous graph by Liu et al. [13], using both the link information and the textual content. They proposed a generative graphical model to find the topic-level direct influence and subsequently the indirect influence between nodes.

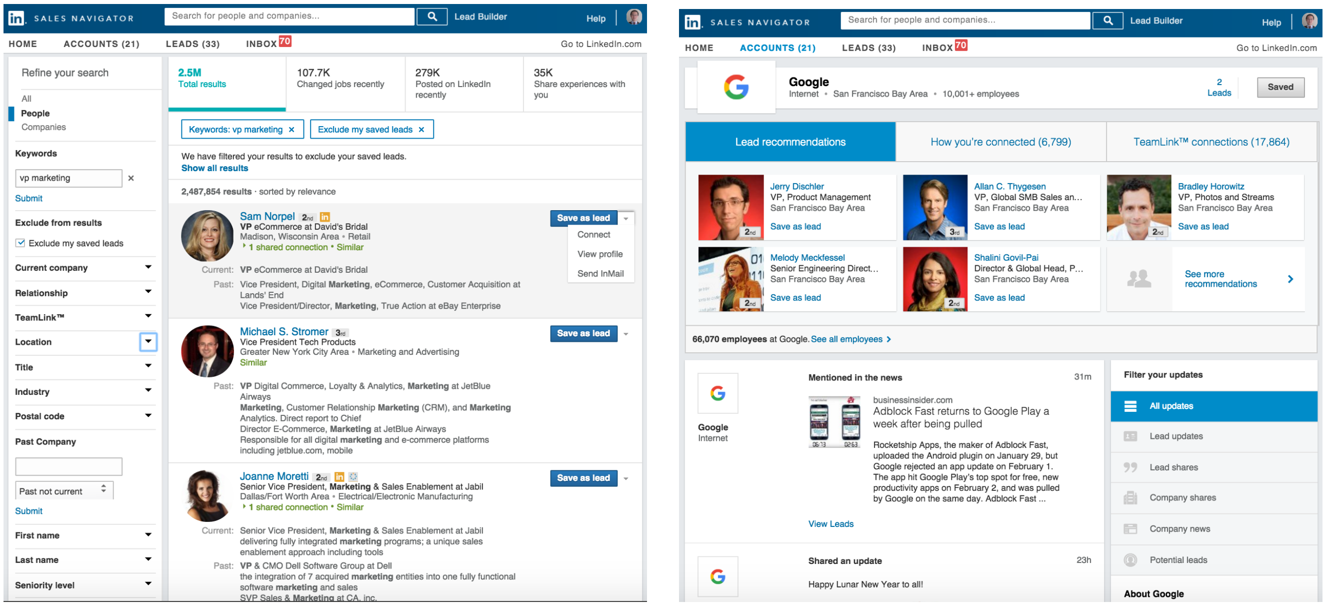
2.4 Semi-supervised label propagation

Another work stream on graphs is to start from a small set of labeled vertices, and leverage the graphical structure to propagate the labels to other unlabeled vertices. The main assumption is that similar vertices are more likely to have the same label. Following these principles, approaches for undirected graphs have been developed [24, 20], as well as a HITS-inspired algorithm for directed graphs [21] and a method motivated by PageRank to identify the unknown labels in a directed graph [22].

2.5 Application to sales use case

Many books and articles have discussed the specific skills a sales professional should have. A crucial one among them is to identify real decision makers and to present their products or services to them [4, 6, 10, 12]. However there has not been many work in the data mining community that aims to help sales professionals tackle this problem (see a recent paper in [5]).

For the decision maker identification problem in our context, we cannot simply run a PageRank or HITS type algorithm to identify the important nodes. Such algorithms typically find influential or popular people on our network (such as celebrities), not necessarily the people who can make sales decisions. We require some labels to guide the algorithms, hence need to combine the labels with the graph structure. Also we will need to deal with heterogeneous graphs within our professional network. With this in mind, the approaches we propose in this paper are able to leverage both the graph information and the contextual information, deal with small amount of labels on the graph, and handle heterogeneous graphs among different types of vertices. We believe they will have positive impact in many sales use cases and other application areas beyond the context we discuss in this paper.



☑ Exclude my saved leads

2.5M Total results | 107.7K Changed jobs recently | 279K Posted on LinkedIn recently | 35K Share experiences with you

Keywords: vp marketing × | Exclude my saved leads ×

We have filtered your results to exclude your saved leads.
Show all results

2,487,854 results · sorted by relevance

Sam Norpel 2nd
VP eCommerce at David's Bridal
Madison, Wisconsin Area • Retail
1 shared connection • Similar
Current: **VP** eCommerce at David's Bridal
Past: Vice President, Digital **Marketing**, eCommerce, Customer Acquisition at Lands' End
Vice President/Director, **Marketing**, True Action at eBay Enterprise

Michael S. Stromer 3rd
Vice President Tech Products
Greater New York City Area • Marketing and Advertising
Similar
Past: **VP** Digital Commerce, Loyalty & Analytics, **Marketing** at JetBlue Airways
Marketing, Customer Relationship **Marketing** (CRM), and **Marketing** Analytics. Direct report to Chief
Director E-Commerce, **Marketing** at JetBlue Airways
Responsible for all digital **marketing** and e-commerce platforms including jetblue.com, mobile

Joanne Moretti 2nd
Senior Vice President, **Marketing** & Sales Enablement at Jabil
Dallas/Fort Worth Area • Electrical/Electronic Manufacturing
1 shared connection • Similar
Current: Senior Vice President, **Marketing** & Sales Enablement at Jabil delivering fully integrated **marketing** programs; a unique sales enablement approach including tools
Past: **VP** & CMO Dell Software Group at Dell
the integration of 7 acquired **marketing** entities into one fully functional software **marketing** and sales
SVP Sales & **Marketing** at CA, inc.

(a) Sales-focused search page

Google
Internet • San Francisco Bay Area • 10,001+ employees

Jerry Dischler 2nd
VP, Product Management
San Francisco Bay Area
Save as lead

Allan C. Thygesen 3rd
VP, Global SMB Sales an...
San Francisco Bay Area
Save as lead

Bradley Horowitz 2nd
VP, Photos and Streams
San Francisco Bay Area
Save as lead

Melody Meckfessel 2nd
Senior Engineering Direct...
San Francisco Bay Area
Save as lead

Shalini Govil-Pai 2nd
Director & Global Head, P...
San Francisco Bay Area
Save as lead

66,070 employees at Google. See all employees ›

Google
Internet

Mentioned in the news 31m
businessinsider.com
Adblock Fast returns to Google Play a week after being pulled
Rocketship Apps, the maker of Adblock Fast, uploaded the Android plugin on January 29, but Google rejected an app update on February 1. The app hit Google Play's top spot for free, new productivity apps on February 2, and was pulled by Google on the same day. Adblock Fast ...

Shared an update 23h
Happy Lunar New Year to all!

(b) Company page with lead recommendation

Figure 1: Screen shots of LinkedIn Sales Navigator, both leveraging LDMS in live production system. (a) shows the refined search page for sales use case, with sales-focused re-ranking of results and other advanced features (spotlight highlights, advanced filters, "save as lead" option for future follow-up). (b) shows the refined company page, with recommended leads on top that are tailored to the specific sales professional user.

3. PROBLEM SETTING

In this section we give a high-level overview of the social network environment, and discuss mechanisms to obtain ground truth and features for developing LDMS. Although the context is within LinkedIn ecosystem, many of the attributes and insights apply to other social network settings.

3.1 Social network environment

LinkedIn is the largest professional social network in the world with more than 400M members to date. Most sales professionals have already started using LinkedIn to research, identify, and reach out to potential clients. With a premium subscription, they can view members' profiles, follow them, get introduced to connect to them, send private messages (called *inMails*) to them, etc.

LinkedIn also has a proprietary product called Sales Navigator, which is designed specifically for sales professionals. It combines LinkedIn's network data, relevant news sources, and the accounts, leads, and preferences set by the sales professionals to produce customized recommendations and insights. See Figure 1 for two screen shots of Sales Navigator on sales-focused search and lead recommendation on company page. Both of them are currently leveraging LDMS in live production systems. We will discuss these specific applications in more detail in Section 6.

3.2 Goal

In this work, we focus on identifying decision makers in the professional social networks within the LinkedIn ecosystem. The decision makers in our context are people who can make or influence a sales decision. Note that we are not interested in influence within the social graph, but rather within the sales context. That's why we are leveraging social data from both the LinkedIn.com and LinkedIn Sales Navigator. Both platforms share certain data elements (e.g., member profiles, connections), but Sales Navigator has its own characteristics (see Figure 1(a)):

- Most users are sales professionals
- On top of LinkedIn.com actions (e.g., view profile, send connection request, send inMails), users can save potential prospects as *leads* for future follow-up
- Users have sales-focused member and company search functionalities

Our goal is to learn a scoring function to assign a LinkedIn Decision Maker Score to each of the 400M+ LinkedIn members. We intentionally make the score *global*, in that every member has *one* score regardless of what his or her industry, company, or function is. In other words, we do not aim to develop a specialized score for members within each industry, each company, or each function. This has broader usage for our context, and can be used within a special setting if needed. One can, for instance, use the score to refine search ranking such that higher-scored members are shown at the top. We can also sort members by this score for a given company to measure decision-making power within the company. Both use cases will be discussed further in Section 6.

3.3 Ground truth

One challenge in the LDMS offline training is the unavailability of the actual ground truth. In other words, we do not have definitive answer on who is and who is not a decision maker. We also do not have access to our customer's CRM (i.e., customer relationship management) data to know exactly who made or influenced a buying decision. Therefore, we seek to use a surrogate ground truth from LinkedIn data. In this section we describe the definition details and the rationale behind it.

3.3.1 The key signal

When a sales professional is interested in a particular individual on LinkedIn, here are the major signals we can collect depending on how he or she expresses interest:

- *Profile View*: view the individual's profile to find out more about the individual,
- *Save as Lead*: within Sales Navigator, save this individual as a lead for future follow-up,
- *Connection Request*: reach out to the individual with a connection request,
- *inMail*: send a direct, private message to the individual.

Going from *Profile View* to *inMail*, the sales professional is getting more serious about this individual. He or she also needs to do more work with the interaction - from simply viewing a LinkedIn profile to carefully drafting a message. Since the *inMail* signal is the most effective way of reaching out through the LinkedIn and Sales Navigator platform, in the LDMS training pipeline we chose the *number of inMails from distinct sales professionals within a specified time frame*, denoted as $N_{in}(x, T)$ for member x in time frame T, to be the signal for ground truth definition. The higher this number, the more likely the recipient is truly a prospect. Note that we only count inMails from one sales professional once for each recipient.

3.3.2 Discount factors

Inbound inMails alone is not sufficient to identify key decision makers for sales use-case as this leads to a lot of false positives. For example, on LinkedIn platform, recruiters also receive high number of inbound inmails. Not surprisingly, they also send out a lot of inMails to prospective candidates. On the contrary, decision makers tend to receive disproportionately more inbound inMails when compared to the amount of inMails that they send out themselves. Additionally, since our motivation is to compute Decision Maker Score across organizations so in our ground truth we discount for organization's popularity amongst sales professionals. Therefore we considered the following discount factors:

- *High-outgoing-inMail discount*:

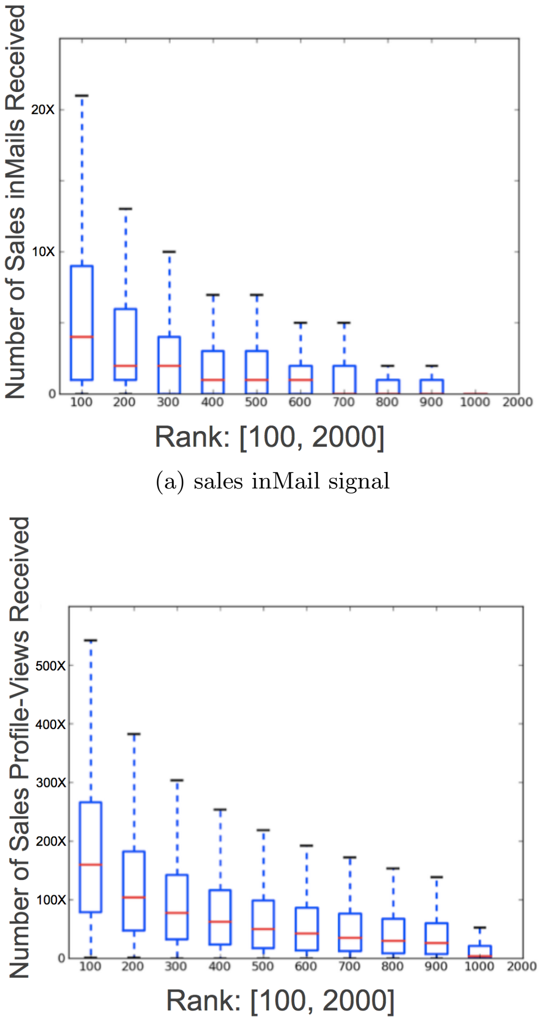
$$D_1(x, T) = \frac{N_{in}(x, T)}{N_{in}(x, T) + N_{out}(x, T) + 1},$$

  to penalize members who send many inMails out (regardless to whom). Here $N_{out}(x, T)$ denotes the number of inMails sent out by member x in time frame T.

- *Peer-comparison discount*:

$$D_2(x, T) = \frac{N_{in}(x, T)}{\max_{y \sim x} N_{in}(y, T) + 1},$$

  to discount members who have some of their peers getting more sales inMails than themselves. Here $y \sim x$ denotes the co-worker relationship (within the same company). The rationale is that the absolute number of sales inMails only makes complete sense within the context of the same company. In other words, if there are peers from the same company getting more sales inMails, the *decision-makerness* of the current member is lower compared to the situation where no one within the same company gets more sales inMails.



(a) sales inMail signal

(b) sales Profile View signal

Figure 2: Sparsity of different graph signals. At the per-company level, we show the number of (a) inMail signals and (b) Profile View signals the top-ranked people (from 100th to 2000th) received from sales professionals within the calendar year of 2015. The box plots averaged all companies with more than 5000 employees based on LinkedIn data. On the y-axis we highlight the scale difference of the two box plots instead of the actual number. It can be seen that Profile View signals are almost 50 folds larger than the inMail signals.

Furthermore, across various time-window we see large changes in the absolute number of inMails that individuals get. Therefore to add robustness to our ground-truth, we consider percentiles within the company for these two discount factors instead of the actual discount values. The final ground truth definition for member x in time frame T is:

$$\mathcal{GT}(x,T) = N_{in}(x,T) * Perc(D_1(x,T)) * Perc(D_2(x,T)),$$

where $Perc$ denotes the percentile which takes value between 0 (0%) and 1 (100%).

3.3.3 Sparsity

One may wonder why not simply use this ground truth definition as the LDMS score (hence no need for supervised learning!). The reason is *data sparsity.* Very few LinkedIn members have ever received an inMail from a sales professional. Figure 2 also shows, at a per-company level, how many inMails and Profile Views top-ranked people received within a 12-month period. It can be seen that other signals, Profile Views in this case, do not suffer as much from sparsity problem. Therefore, to robustly rank a broader set of member base, we need to identify other non-sparse signals that correlate with the ground truth.

3.4 Features

In the LDMS offline training pipeline we considered various information about the member and the social graphs from LinkedIn.com and Sales Navigator. The following are the high-level categories:

- *Member Profile*: title, position, seniority, and related working experience of the member
- *Connection Graph*: an undirected member graph in which each edge is a first-degree connection between two members
- *Invitation Graph*: a directed member graph in which each edge is a connection invite action
- *Profile View Graph*: a directed member graph in which each edge is a profile view action
- *Lead Save Graph*: a directed bipartite graph between members and sales professionals in which each edge is a lead save action in Sales Navigator
- *inMail Graph*: a directed member graph in which each edge means an inMail was sent

All graph edges have a time stamp associated with it, and for some graphs there might be multiple edges between two nodes (e.g. *Profile View Graph*). There are other social graphs within the LinkedIn ecosystem, such as the Follower Graph, Endorsement Graph, etc. We currently do not use them but the framework is easy to extend to these additional graphs. Also note that we make an explicit distinction among various graphs on LinkedIn instead of blending all the actions into one graph. In Section 5.3 we will explore why this is helpful. We also want to emphasize that these heterogeneous graphs cover different types of graphs we normally see in social networks and include undirected, uni-directional and bipartitie graphs.

4. LEARNING APPROACH

Given the aforementioned ground truth and feature groups, in this section we introduce two algorithms for LDMS training, mainly based on how we encode the graph information into the learning process. The graph summarization approach summarizes each graph into specific features, and then adapts state-of-the-art supervised learning algorithms. The bipartite graph learning approach leverages the graphical structure explicitly and presents a novel graph propagation algorithm. Though we focus our algorithmic descriptions within the LinkedIn context, the approaches are general and should be applicable to other applications.

4.1 Learning with Graph Summarization

In this approach we cast the problem to a classification problem, i.e., we are training a classifier to categorize each member to be a decision maker (+1) or not (-1). The key here is how to leverage various social graphs for classifier training on top of the member-profile-based features. In the following we discuss the details for three different types of graphs. Since all the graphs evolve with time, we fix a specific time frame for each graph.

For undirected graphs (e.g., *Connection Graph*), we extract the following information from the graph for each member:

- *All-Degree*: Degree of the member in the graph,
- *Sales-Degree*: Degree of the member considering only neighbors who are sales professionals,
- *Ratio-Sales-All*: The ratio of *Sales-Degree* and *All-Degree*.

For non-bipartite, directed graphs (e.g., *Profile View Graph*, *Invitation Graph*, *inMail Graph*), we compute the following information from the graph for each member:

- *All-In*: Indegree from all members,
- *All-Out*: Outdegree to all members,
- *Sales-In*: Indegree from sales professionals,
- *Ratio-Sales-In*: The ratio of *Sales-In* and *All-In*,
- *Ratio-In-Out*: The ratio of *All-In* and *All-Out.*

For the bipartite directed graph (e.g., *Lead Saves*) between members and sales professionals, we only compute the *Sales-In* feature.

One key insight from this graph summarization approach is that the ratio features are more relevant to distinguish good and not-so-good decision makers, as good decision makers tend to have more in-bound interest (which can be profile views, lead saves, inMails, etc.) and less out-bound signals.

For supervised learning, we created binary labels based on the ground truth values, and combined all the graph-based features with the member-profile-based features. We can use any classification algorithm for training – for comparison in this paper we used elastic net classifier [25], which is a combination of L2 and L1 regularization. From our experience it was shown to be more flexible and can strike the right balance between goodness-of-fit and sparsity. One may also adapt Learning-To-Rank methods to learn the overall ranking of the members based on LDMS ground truth.

4.2 Bipartite Graph Learning

One issue the graph summarization approach does not take into account is the *quality* or *competency* of sales professionals. In the graph features computed from each of the graphs, we effectively weigh each sales professional equally. This may not be ideal as the signal, say a Profile View, from a novice sales professional should carry much less weight compared to that from an experienced and active sales professional. In this subsection we formulate this as a bipartite graph learning problem which explicitly takes into account the LDMS for a member and the competency assessment for a sales professional, which we call the *LinkedIn Sales Competency Score* (LSCS). The motivation is that when we compute the LDMS score for a member, the incoming signals from the sales professionals should be weighted by their respective LSCS score. Likewise we can also take into account members' LDMS score when updating the LSCS score for a sales professional.

Formally, let $\mathcal{X} = \{x_i\}$ and $\mathcal{Z} = \{z_j\}$ be two disjoint sets of nodes in the network. With a slight abuse of notation we use x_i and z_j to denote both the nodes in the network as well as the contextual feature vectors they represent. We assume there are K bipartite graphs between nodes $\mathcal{X}$ and $\mathcal{Z}$. For the k-th graph $G_k = (V_k, E_k)$, we have vertices $V_k = \mathcal{X} \cup \mathcal{Z}$, and edges $E_k = \{e_{k(ij)} : x_i \in \mathcal{X}, z_j \in \mathcal{Z}\}$. We allow the edges to be undirected ($e_{k(ij)}$), uni-directional ($e_{k(i \to j)}$), or bi-directional ($e_{k(i \leftrightarrow j)}$). In the following we use the uni-directional edges to describe the approach but it can take all three types of edges.

For the bipartite graph construction we assume $\mathcal{X} \cap \mathcal{Z} = \emptyset$. In our context $\mathcal{X}$ can be the members, and $\mathcal{Z}$ the sales professionals. Since sales professionals are also members, in our context they have a representation on both sides.

Let p_i be the p-score for x_i, and q_j be the q-score for z_j. These, in our context, can be LDMS for members and LSCS for sales professionals. We define the following models for p_i and q_j:

$$p_i = f\left(w_{pc} x_i + \sum_{k=1}^{K} w_{pk} \sum_{j: e_{k(j \to i)} \in E_k} q_j \cdot t(j,i) \right), \quad (1)$$

$$q_j = g\left(w_{qc} z_j + \sum_{k=1}^{K} w_{qk} \sum_{i: e_{k(i \to j)} \in E_k} p_i \cdot t(i,j) \right). \quad (2)$$

Here $f(\cdot)$ and $g(\cdot)$ denote non-linear transformation functions from the continuous space to $[0, 1]$. We assume they are both sigmoid functions in this paper. w_{pc} and w_{qc} are the weight vectors for all constant (individual) features of x_i and z_j that are not based on the bipartite graphs. In our context these features include profile-based features such as title, position and seniority. Note that they are multi-dimensional and can be of different length for x and z. w_{pk} and w_{qk} are the weights for graph G_k and they measure how graph G_k contributes to the overall model regardless of index i or j. $t(j, i)$ is a pre-defined transformation function for pairs (z_j, x_i). In the simplest case it can be the count of signals z_j sent to x_i in the given time frame.

This model can be seen as a generalization of many existing supervised learning and graph-based methods. With $q_j \equiv 1$, the model goes back to standard supervised learning with summarization features from the bipartite graphs, as we discussed in Section 4.1. If $K = 1$ and we have no constant features, i.e., $x_i \equiv z_j \equiv 0$, the model is an extension of the label propagation approach [24], where any label (of p_i) is first propagated to the other side of the bipartite graph (to q_j's), and then propagated back (to update all p_i's). If we remove the ground truth labels and make this an unsurprised learning problem, the model is an extension of the well-known HITS algorithm [11] to bipartite graphs, in which p_i and q_j can be thought of as the authority and hub scores, respectively.

For learning, we are given ground truth labels for the p-score, and need to estimate $\mathcal{W} = \{w_{pc}, \{w_{pk}\}, w_{qc}, \{w_{qk}\}\}$ (with parameter regularizations). Instead of directly optimizing the problem which is non-convex, in this paper we present an iterative approach and introduce an intertwined elastic net solver to solve the problem. The iterative algorithm is shown in Algorithm 1. As a starting point we initialize the q-score by fixing $w_{qc} \equiv w_{qk} \equiv 1$. Then we solve an optimization problem for w_{pc} and w_{pk} by fixing q_j scores. This can be done exactly the same way as what we discussed in Section 4.1. After that we propagate p_i and q_j scores throughout the entire network by iterating equations (2) and (1), with fixed model parameters $\mathcal{W}$. Note that the unlabeled x_i's within $\mathcal{X}$ will also get propagated p_i scores. We iterate until we reach a stable state of p_i and q_j scores across the entire network. Then we repeat the process of optimizing p_i and q_j separately for the respective parameters, and propagating the labels through the network with the learned parameters. In Step 6 where we optimize the q_j scores, since we do not have ground truth labels for q_j's, we take a heuristic approach and label the top 20% percentile as $+1$ and bottom 20% percentile as -1. From our experiments we see the results and convergence rate are not sensitive to the specific percentile choice. The algorithm typically converges within 20 iterations from our experience.

Algorithm 1 The Bipartite Graph Learning algorithm

```
1: Initialization: $w_{pc} = w_{qc} = 1, w_{pk} = w_{qk} = 1, \forall k$.
2: Compute $q_j$ using (2), $\forall j$.
3: (Optimization) Solve elastic net problem in (1) for $w_{pc}$ and $w_{pk}, \forall k$.
4: (Propagation) Iterate (2) and (1) till convergence with fixed $\mathcal{W}$, obtaining $p_i$ and $q_j$, $\forall i, j$.
5: repeat
6:     (Optimization) Solve (2) for $w_{qc}$ and $w_{qk}, \forall k$.
7:     (Optimization) Solve (1) for $w_{pc}$ and $w_{pk}, \forall k$.
8:     (Propagation) Iterate (2) and (1) till convergence with fixed $\mathcal{W}$, obtaining $p_i$ and $q_j$, $\forall i, j$.
9: until convergence
```

5. METHODOLOGY AND RESULTS

For the results we show in this paper, we collected all LinkedIn network data over the calendar year of 2015. The ground truth was defined based on Section 3.3 with this time frame, and all the feature groups defined in Section 3.4 were captured for this time frame. There were around 400M members in the pipeline, and the total number of signals (e.g., profile views) for most of the graphs is in the hundreds of millions. We leveraged our internal Hadoop infrastructure for distributed learning.

5.1 Offline Training Methodology

Table 1: NDCG results for graph summarization (Summarization) and bipartite graph learning (Bipartite)

NDCG@K	10	20	50	100	500	1,000	5,000	10,000	50,000	100,000	500,000	1,000,000
Summarization	**1**	0.963	0.9084	0.8593	0.9039	0.8684	0.8682	0.7987	0.8339	0.8701	0.9336	0.9344
Bipartite	**1**	**0.9664**	**0.9665**	**0.9063**	**0.9183**	**0.878**	**0.871**	**0.8043**	**0.8412**	**0.8778**	**0.9367**	**0.9373**

Table 2: Kendall's τ results for graph summarization (Summarization) and bipartite graph learning (Bipartite)

Kendall's τ@K	10	20	50	100	500	1,000	5,000	10,000
Summarization	**0.5394**	**0.5769**	**0.5185**	**0.6365**	**0.5681**	0.4829	0.4717	0.4956
Bipartite	0.4045	0.4476	0.5135	0.6253	0.5605	**0.4855**	**0.4746**	**0.5043**

For offline evaluation we randomly split the LinkedIn member base into training (70%) and testing (30%). Based on the ground truth definition in Section 3.3, we assign a (+1/0/-1) label to each member based on two pre-selected thresholds T_+ and T_-:

$$\mathcal{L}(x,T) = \begin{cases} +1, & \text{if } \mathcal{GT}(x,T) \geq T_+, \\ -1, & \text{if } \mathcal{GT}(x,T) < T_-, \\ 0, & \text{otherwise.} \end{cases}$$

For the graph summarization approach, we ignored the members who have label 0 to remove noisy data. For bipartite graph learning approach, we only used members with labels $(+1/-1)$ in each round of elastic net training for LDMS (Step 7 in Algorithm 1), but let the labels propagate to the other members throughout the iterations (in all propagation steps).

5.1.1 Metrics

For offline evaluation we need to compare the ranked list of decision makers from the algorithm with that from the ground truth definition. We are more interested in the relative ranking of members rather than the absolute predictive scores. Therefore we chose NDCG (normalized discounted cumulative gain) and Kendall's τ with our own adaptions.

NDCG is widely used in the Learning-To-Rank literature [1, 3] to measure ranking performance. For a ranked list and position k, $NDCG@k$ is the ratio of $DCG@k$ to the ideal $DCG@k$, where the latter is obtained had the list been sorted by the ground truth label. $DCG@k$ is formally given by:

$$DCG@k = \sum_{r=1}^{k} \frac{2^{rel(r)} - 1}{log_2(r+1)}, \tag{3}$$

which captures the importance of finding the correct ordering among the top ranked decision makers. Since the ranked list is very long in our case, we chose various k position from 10 all the way to $1,000,000$. Note that we have only one very long ranked list to evaluate, unlike the typical search scenario that multiple search sessions contribute to the NDCG metric. The gain function $rel(r)$ is defined by binning the ground truth label into different buckets.

Kendall's τ is a standard way of measuring correlation, which has also been extensively used in the literature of identifying the most influential people [15, 16, 19] in order to compare the ranking lists of different algorithms. Formally, it is defined as:

$$\tau(k) = \frac{(\#\text{concordant pairs}) - (\#\text{discordant pairs})}{\frac{1}{2}k(k-1)}$$

at position k, where $\#pairs$ represents the number of pairs. A pair of members are *concordant* if they are ranked the same way in the predictive algorithm and the ground truth, and are *discordant* if they are not. Note that unlike NDCG which is between 0 and 1, $\tau(k)$ is between -1 (perfect disagreement) and 1 (perfect agreement). One key difference of Kendall's τ from NDCG is that in Kendall's τ each member pair has the same weight as opposed to a discounted weighting scheme in NDCG. In our experiments we compute $\tau(k)$ for k between 10 and 10,000.

5.2 Overall Results

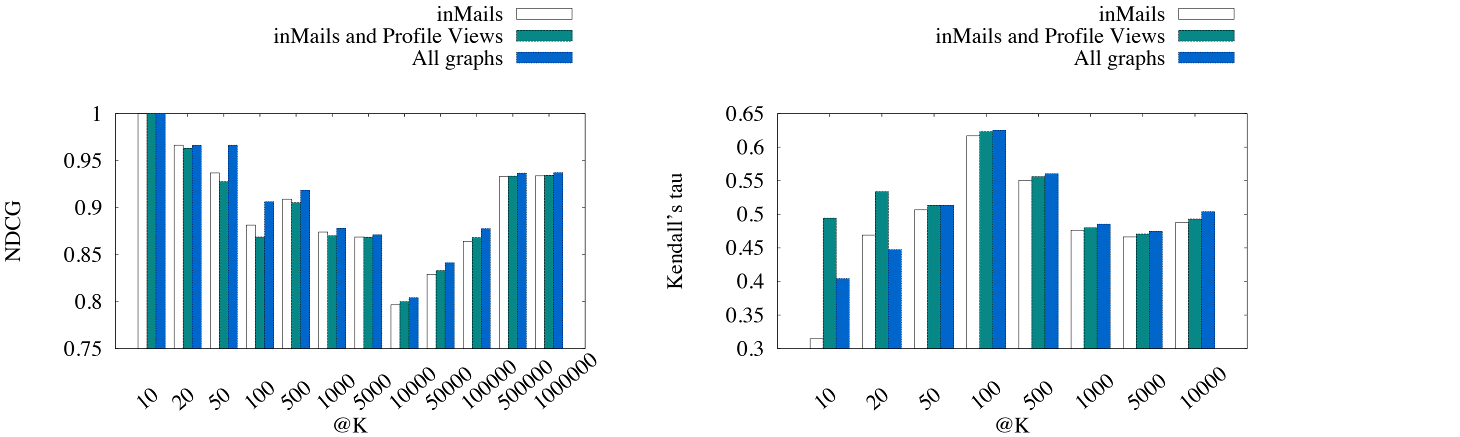
The performance of the two approaches described in Sections 4.1 and 4.2 is shown in Table 1 and Table 2. First of all, both approaches performed very well for the LDMS ranking problem. Between these two methods, we can see that the bipartite graph learning approach outperformed the graph summarization approach in terms of NDCG for all list sizes. For Kendall's τ, the results show that the graph summarization approach had better Kendall's τ than the bipartite graph learning method for small list sizes $(10-500)$, but was inferior to bipartite graph learning method for large list sizes $(1,000-10,000)$. As we are more interested in how the model would perform with a reasonably large amount of member base for the decision maker ranking, the bipartite graph learning approach is superior overall. Also keep in mind that even a small improvement in these two metrics will typically lead to significant improvement in downstream applications. We will cover one such case in Section 6.

One reason why the improvement tended to be small is that the inMail graph features are more important from both approaches due to the fact that the ground truth definition also leveraged the inMail information. The additional improvements over the graph summarization approach from the bipartite graph learning approach is significant in that it shows a bootstrap from the basic graph summaries with the help of the LSCS score for the sales professionals will lead to a better ranking result overall. We also get the side-benefit of having a ranking among the sales professionals, which by itself has many downstream applications. The details of this are beyond the scope of this paper.

5.3 Leveraging Different Social Graphs

In Tables 1 and 2, we used all the social graphs we have introduced in Section 3.4. In this subsection we evaluate how much each social graph contributes to the overall performance on top of the inMail graph. In Figure 3, we compare NDCG and Kendall's τ results for the following graph configurations:

- using only the inMail graph



(a) NDCG

(b) Kendall's τ

Figure 3: NDCG and Kendall τ results for bipartite graph learning approach using different graphs

- using the inMail and Profile View graph
- using all social graphs

We can see that in the majority of the cases the addition of more graphs helps improve the performance when K is reasonably large. This agrees with our intuition that considering more social graphs leads to a better ranking of LDMS among decision makers, and also stronger signal for the strength of the relationship between a decision maker and a sales professional. The additional social graphs not only resolved the sparsity problem we mentioned in Section 3.3.3, but also improved the overall performance.

5.4 Score Distributions

Last but certainly not the least, we plot the histograms of the LDMS score for members and LSCS scores for sales professionals in Figures 4 for the bipartite graph learning approach. The LDMS score for the graph summarization approach follow a similar pattern. We want to highlight that the two scores behave in a totally different way. LDMS scores follow a power law distribution, where the majority of people have scores close to 0. This agrees with our intuition that decision makers are a small percentages of the member base. On the other hand, LSCS scores follow a normal distribution, with the majority of sales professionals around 0.5. This indicates that the majority of sales professionals perform neutrally, with some distinguished and some not very experienced. This is also intuitive and can be supported by various reports [4, 6].

6. APPLICATIONS

One key component of social prospecting is to identify people with influence in the sales process. This is one of the core components of the LinkedIn Sales Navigator product. We discuss some of the applications of LDMS in this section, all of which are currently running live on the Sales Navigator product. On the offline side, both of the aforementioned offline training algorithms are computed weekly on our internal Hadoop infrastructure. Once it is completed, both versions of the LDMS scores are pushed to a key-value store for online consumption.

6.1 Sales Navigator Search

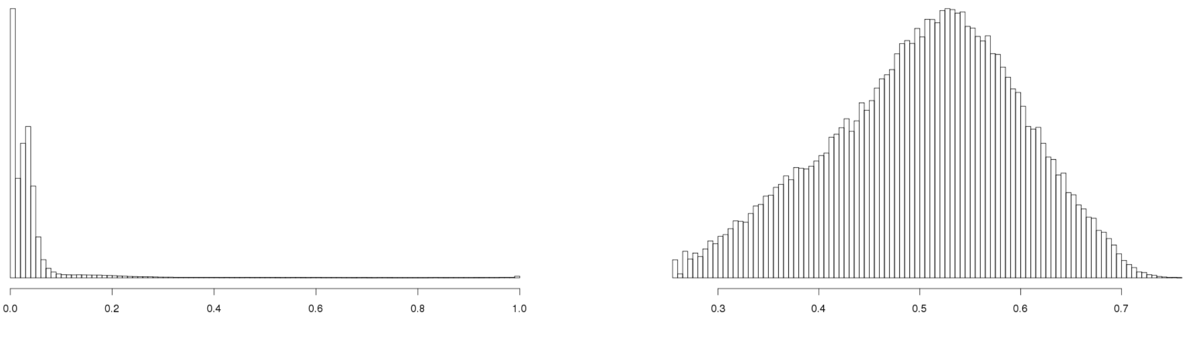
Search is one of the key building blocks of Sales Navigator. We have integrated the LDMS score into Sales Navigator Search to allow the users to find decision makers faster, which differentiates Sales Navigator search to the LinkedIn.com search (see Figure 1(a)). We first investigated the integration performance offline using the NDCG metric, for which we wanted to test whether the LDMS-enabled search ranking is better than the baseline model (which was already fine-tuned with years of effort). Note this is the NDCG metric on search rankings, not on LDMS ranking as we presented in the previous section. The baseline NDCG performance was 67.3%, LDMS score from the graph summarization approach yielded 68.6%, and the LDMS from the bipartite graph learning approach yielded 69.8%. This seemingly small improvement on offline NDCG turns out to have a significant impact on online search metrics. Compared to the baseline search metrics, the A/B test for the graph summarization approach has shown 8.4% improvement on lead saves from search and 3.7% improvement on inMail sent, which are key metrics for Sales Navigator search. We are currently conducting A/B test for the bipartite graph learning approach and are expecting to see more improvements on these key metrics.

6.2 Lead Recommendation

Lead recommendation module in Sales Navigator recommends potential leads to the user based on the decision making power those individuals have and the user's past activities on LinkedIn and Sales Navigator, such as profile views, lead saves, and the specified sales preferences. See Figure 1(b) for lead recommendation module on the company page. In the simplest implementation, it is extracted from the top decision makers for the specific company and then filtered by the users activities and preferences. This is a new module that was recently launched, and was made available because of the LDMS score. So far we are seeing this module has the largest user engagement on the company page.

6.3 Decision Maker Insights

Another LDMS application is on the Sales Navigator In-



(a) LDMS scores

(b) LSCS scores

Figure 4: Histogram of the (a) LDMS scores for members and (b) LSCS scores for sales professionals

sights module, which is a feed system that aims to keep the users informed about latest updates on leads and companies they care about. This includs job changes, article shares, mentions in the news, company updates, etc. LDMS is leveraged to boost rankings of those updates from key decision makers.

7. CONCLUSION

In this paper we presented LDMS, the LinkedIn Decision Maker Score, to capture the ability to make or influence a buying decision for each of the 400M+ LinkedIn member. We proposed two learning approaches to tackle this problem, which are able to leverage both the graph information and the contextual information on the vertices, deal with small amount of labels on the graph, and handle heterogeneous graphs among different types of vertices. Although the approaches were presented in the LinkedIn context, they are general methods and can be applied to other social network settings.

Within LinkedIn Sales Navigator, this score is used in refining search ranking, improving lead recommendations and providing additional insights about decision maker activity on LinkedIn. More broadly for LinkedIn, we will build upon this work to do more holistic optimization to improve overall LinkedIn experience for key decision makers across all enterprise products.

8. REFERENCES

[1] S. Boyd, C. Cortes, M. Mohri, and A. Radovanovic. Accuracy at the top. In *Advances in neural information processing systems*, pages 953–961, 2012.

[2] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine, 1998. In *Proceedings of the Seventh World Wide Web Conference*, 2007.

[3] O. Chapelle and Y. Chang. Yahoo! learning to rank challenge overview. In *Yahoo! Learning to Rank Challenge*, pages 1–24, 2011.

[4] T. Connor. *91 Mistakes Smart Salespeople Make.* Sourcebooks, Inc., 2006.

[5] B. A. Duncan and C. P. Elkan. Probabilistic modeling of a sales funnel to prioritize leads. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1751–1758. ACM, 2015.

[6] J. J. Fox. *Secrets of Great Rainmakers: The Keys to Success and Wealth.* Hachette Books, 2006.

[7] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with trustrank. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 576–587. VLDB Endowment, 2004.

[8] T. Haveliwala, S. Kamvar, and G. Jeh. An analytical comparison of approaches to personalizing pagerank. 2003.

[9] T. H. Haveliwala. Topic-sensitive pagerank. In *Proceedings of the 11th international conference on World Wide Web*, pages 517–526. ACM, 2002.

[10] A. B. Karl Schmidt, Brent Adamson. Making the consensus sale. `https://hbr.org/2015/03/making-the-consensus-sale`, 2015. [Online].

[11] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[12] D. Kurlan. Top 4 reasons salespeople struggle to reach decision makers. `https://www.salesforce.com/blog/2014/03/sales-decision-makers-gp.html`, 2014. [Online].

[13] L. Liu, J. Tang, J. Han, M. Jiang, and S. Yang. Mining topic-level influence in heterogeneous networks. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 199–208. ACM, 2010.

[14] Y. Liu, B. Gao, T.-Y. Liu, Y. Zhang, Z. Ma, S. He, and H. Li. Browserank: letting web users vote for page importance. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 451–458. ACM, 2008.

[15] Q. Meng and P. J. Kennedy. Discovering influential authors in heterogeneous academic networks by a co-ranking method. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages

1029–1036. ACM, 2013.

[16] A. Pal and S. Counts. Identifying topical authorities in microblogs. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 45–54. ACM, 2011.

[17] M. Richardson and P. Domingos. The intelligent surfer: Probabilistic combination of link and content information in pagerank. In *NIPS*, pages 1441–1448, 2001.

[18] D. Saez-Trumper, G. Comarela, V. Almeida, R. Baeza-Yates, and F. Benevenuto. Finding trendsetters in information networks. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1014–1022. ACM, 2012.

[19] J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 261–270. ACM, 2010.

[20] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. *Advances in neural information processing systems*, 16(16):321–328, 2004.

[21] D. Zhou, T. Hofmann, and B. Schölkopf. Semi-supervised learning on directed graphs. In *Advances in neural information processing systems*, pages 1633–1640, 2004.

[22] D. Zhou, J. Huang, and B. Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of the 22nd international conference on Machine learning*, pages 1036–1043. ACM, 2005.

[23] D. Zhou, S. Orshanskiy, H. Zha, C. L. Giles, et al. Co-ranking authors and documents in a heterogeneous network. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 739–744. IEEE, 2007.

[24] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the Twentieth International Conference on Machine Learning (IC ML-2003)*, 2003.

[25] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.