What Determines Your Promotion? A Novel Exploration of Civil Careers in Ancient China Using Network Analysis

Jingjing Tian
Department of Computer Science
Stanford University
jtian97@stanford.edu *

Yanbang Wang
Department of Computer Science
Stanford University
ywangdr@stanford.edu *

1. INTRODUCTION

What determines political promotion of civil servants in ancient China's feudal system? This used to be a central question studied by historians and archaeologist. We are the first to leverage the newly released China Government Employee Database-Qing (CGED-Q) recently [17] to investigate this from a novel but promising perspective. Central to the project are three problems concretizing our research theme: career-high position ranking prediction, "stepping-stone" positions mining, and analysis and modeling of career graph's evolution. Our work shows that studying these problems with various relational graphs built from the database can yield highly fruitful result: we manage to verify an unofficially documented phenomenon of promotion, which says that civil servants sharing certain common characteristics were very likely to form latent allies and help each other in career movement. We also design a graph algorithm that identifies a group of special, latent positions that were long off historical records. The methods that we propose has several advantages over traditional research on history, including being much more quantitative and rigorous in problem formulation, as well as being able to deriving conclusions from more universal aspects.

The rest of the report is structured as follows: First, we give a brief review of previous research that is related to our work. Then, we will formulate our central problems of interest. We proceed by illustrating how we deploy the tools in network science for comprehensive analysis in the employment data in history and how we derive interesting conclusions through some experiments and analysis. For a more thorough understanding, we recommend readers go through some background information we've provided, including outline of structure of organization of Chinese feudal government at that time, as well as complexity of career movement in political system.[1]

2. RELATED WORK

In this part, we first give a brief introduction on previous research on career transition from the perspective of sociology and history, which gives us certain domain knowledge and inspiration for subsequent formulations. Then we conduct literature review on the three sub-tasks, which covers a wide range of perspectives, as well as how our work is related to those papers.

2.1 Analysis of Career Transition

From the perspective of sociology and history, there has been large amount of research conducted on the personnel career transition across different time periods and regions. Li et al. [9] first studied the personnel career transition in Qing Dynasty quantitatively, which is why we base our project on this paper as social background. In this paper, the authors focus on the impact of birth origin of individuals on their career paths. Based on the materials collected from employment data, the authors conducted quantitative analysis on numerous records in the history and discovered some interesting patterns. The amount of data and quantitative methods used in their work pioneered in this field. However, one key weakness of their work is that their work is majorly limited to the discussions on some classification and data mining from simple patterns in the database, leaving more interesting and complicated patterns to be discovered. **Our work could be related to their work as an extension on the topics, but with more advanced techniques.**

2.2 Network Analysis and Modeling

There also has been tremendous analysis on evolving patterns of networks. In [1] authors study scientific papers spanning a long time period. They effectively visualized the interesting patterns they've discovered in citation and author collaboration. Another example is [5] where authors study the evolution of roles in reddit networks. These papers provide insights in temporal analysis and effective visualizations. Newman [11] gave a comprehensive investigation into important properties including clustering coefficient, degree distribution and reachability. **These works relate to our work in that they provide us with some basic perspectives to view the properties in networks.**

On the other hand, there has been a trend of research focusing on modeling and simulating the creation, evolution and behaviors of realistic networks. Such kind of endeavors are useful in understanding the inner mechanisms and rules of evolution of networks. In Newman's work [11], small world model as well some other random models are introduced, along with how they satisfy the attributes of realistic networks. Leskovec et al. [8] introduced two models, Community Guided Attachment and Forest Fire Model to explain

*Equal contribution

[1]We put this part in the shared doc, which we recommend you go through as well. The same contents is included in the Milestone. https://docs.google.com/document/d/1xdEnYbobAf0sUzRnENpJoVdqDAAlH0tjEwQDksrmv4A/edit?usp=sharing

evolving networks that have long-tailed degree distribution and shrinking diameters, of which the latter fit the properties well, while the former fails on shrinking diameters. **Our work is related to such work in that these models give us good starting point, while still leaving some room for us in the specific settings.**

2.3 Centrality and Ranking

As mentioned in 1, our task of discovering stepping stone positions can be regarded as problem of ranking or centrality, with stepping stone positions ranked higher and otherwise lower.

There have been tremendous efforts on calculating the preminence of nodes in networks and identifying important objects. In [14], Sun et al. gave an overview of the endeavors on tackling the ranking problems. Overall the idea lies in that the centrality of a node is determined by its interaction between other nodes in the network. Some useful metrics include degree centrality [12], where centrality is simply defined as the degree of a node, sometimes in normalized form, betweenness centrality[4], where we consider how a node matters in the shortest path of pairs of other nodes, and so on. Techniques for ranking and calculating centrality have been applied successful in many fields. One example is PageRank [13], where pages are regarded as nodes, and links between them are used for centrality calculation. Another is HITS [7] which provides another perspective on utilizing information in links, identifying authorities and hubs in the network. Based on the intuitive ideas, many variants have been proposed to meet specific needs regarding link weights and personalized ranking. **Generally speaking, our endeavors in discovering SSP discovery is related to this field of study as an application of ranking, and finding nodes with high centrality, where the definition is tailored in the realistic setting.**

2.4 (Semi-)supervised Prediction

Lying at the core of machine learning, classification and regression are two most significant tasks that we pay attention to. **Naturally, our goal of predicting the career of an individual can be formulated intuitively as predicting the level/pinji of the last position, which falls in the scope of regression.** Traditional prediction pipeline includes extracting features as well training models such as SVM, decision trees for fitting and predicting.

As we'll propose later, semi-supervised classification on graphs exhibits powerful performance on structured data. One important stream falls in graph regularization. Classic methods include label propagation [18], where authors propagate the label information from labeled nodes to unlabeled nodes. Another method in this genre is manifold regularization [2], where geometrically nearby data points are enforced similar labels. Also proved effective is the method called iterative classification [10]. The algorithm iterates between updating features using adjacent neighbors' information and updating labels according to new features. All these methods require careful construction of graphs.

We omitted the deep learning techniques here due to low relevance.

3. PROBLEM DEFINITION

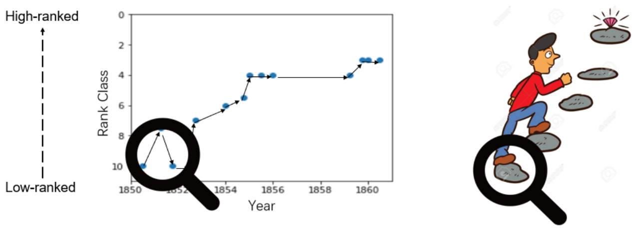


Figure 1: Illustration of Stepping-stone Positions

We carefully frame three tasks to perform on the dataset, in order to gain insight into the dataset from three different angles. Solving each requires us to develop and adopt a different set of techniques of network analysis.

3.1 Task 1: Network Analysis and Modeling

There are many ways to construct networks from the CGED-Q dataset, which exhibit temporal properties that leave us to explore. Given a certain collection of records, we could safely construct a network as described in 4.2. Static analysis includes diameter, largest connected component, in/out degree distribution and clustering coefficients. One interesting aspect to investigate here is that if we assume the records arrive in order according to their timestamps, the network forms in a growing fashion and in this way, we get a 'series' of graphs, denoted as $G_1, G_2, \ldots, G_M$, where m is the number of records. Suppose f is a function that calculates a specific property of graphs, we get series of properties of any angle, denoted as $f(G_1), f(G_2), \ldots, f(G_M)$. Are there interesting patterns in such networks? Which properties evolve while others remain stable? There questions are left for us to explore. Another important aspect lies in modeling the inner mechanisms of formulation of the networks. We should provide a model M that given N as number of nodes, our model should generate a graph that has similar attributes similar to realistic career networks. By the word similar, we are refering to the essential properties, such as number of edges, in/out degree distributions, diameter, clustering coefficient, largest connected component and so on. For more information on network modeling, refer to 2.2. We will elaborate our formulations of different aspects of the analysis in 5.1.

3.2 Task 2: Step-stone Position Discovery

Stepping-stone positions refer to a set of special positions – they are special in the sense that, throughout history, people served in those positions seemed to gain promotion much more quickly than others. However, most of them were absent from official administration records, and were often only found in anecdotes. It was believed that the ruler internally kept a list of such important and "good" positions. However, no effort was ever made to identify them on a large scale by historians.

Therefore, it remains a question as to whether we could find the most possible stepping-stone positions in the aforementioned career network. It also deserves careful consideration as to what information from the network could be utilized for this task.

A more formal definition is the following. First, we follow [15] to define a *career trajectory*: we can represent the trajectory of a person's career development as follows: $p_1 \rightarrow t_1 \rightarrow p_2 \rightarrow t_2 \rightarrow \cdots \rightarrow t_{n-1} \rightarrow p_n$ where p_i corresponds to the i-th position the person took chronologically, and t_i corresponds to length of time he spent in position p_i before being nominated to another position. n is the number of positions of each career trajectory. Note that p_i and p_{i+1} are likely to be the same. We assign a 'level', or so-called *pinji*, which means the hierarchical level of the position in the feudal system of Qing, determined from the title of the position. There were a total of 19 different pinji's from the lowest at level-10 all the way up to the highest at level-1, with an interval of 0.5. We call a position p_i, a *stepping-stone position*, if people who's taken this position have a higher chance to migrate to a higher pinji in the future. We use potential to denote the possibility, or our confidence in that a position is a stepping-stone. In our ranking method, potential could also be related to the ranking score. Our task is to find stepping-stones with high possibility, validated by some evidence and case studies.

3.3 Task 3: Prediction of Career Achievement

Though measuring a person's career achievement can be arguably complex, here we adopt "ranking of a person's last position" in his political career as the most straightforward measurement. We propose the following three questions based on this:

- **Q1**: Knowing all the person's demographic attributes (e.g., family background, civil entrance test score, dynasty, etc.) and information about his first job appointment, (how accurate) can we predict ranking of the person's *last* position in his career?
- **Q2**: How can we using (latent) relational information among people to increase our model's accuracy? For example, knowing one person's "career achievement" ground truth, how much does it help for us to predict the other person's "career achievement" if they were born in the same place?
- **Q3**: Based on last question, if we turn out finding that multiple types of relational information are useful for modeling, which ones are the most important, and how can we combine to make even better prediction?

These questions are heavily inspired by several piece of historical research that suggest similarity and parallelism of career trajectories may happen among people sharing certain attributes[6; 16]. In **Q1** we aim to investigate how much can a person's attributes on and before his entrance to civil system would affect his career movement many years later. Our expectation is that answering this question would be a strong test against many previous conjectures about the mobility of officers in the feudal system. **Q2** serves two purposes: it not only provides a possible direction to achieve better performance in **Q1**, but also reveals how important the many latent relationships (personally, geographically, temporally, etc.) among people may strongly affect their career movement over many years. **Q3** further investigate and compare strength of these effects.

We introduce a more formal notation of the problem is defined as follows: the appointment records of a person i can be represented as a list $\{X_{i.}\} : X_{i_1}, X_{i_2}, \ldots, X_{i_m}$. m is the total number of records of that person. Records' indices are sorted in ascending order of timestamp associated with each appointment record. Besides the list of records, each person also has a label Y_i that represents the *last* position's ranking in his career, i.e., the position ranking associated with X_{i_m}. Y_i is a real number with discrete values, ranging from 1 to 10, with a gap of 0.5 between each two adjacent levels. Therefore, there are 19 possible values for Y_i. Our goal is to predict the Y_i using only two sources of information: 1) anything in X_{i_1}, 2) Other general background properties of the person that do not appear in appointment records. Note that the label (ground truth) is sometimes also called textitpinji in our terminology.

Examples of general background properties of the person of a person family origins, name, birthplace, dynasty that he lived in, etc.

4. DATA PREPROCESSING

4.1 Dataset description

Our data is obtained from the website of Lee-Campbell Research Group [3] [2]. The most updated database comprises of about 3.3 million records. Each record (row) correspond to one government employer taking up one specific position during a certain time period. The records are timestamped quarterly (four times annually), indicating that we could study it from the aspect of time evolution. There are over 90 column variables in the dataset, describing in great detail information about the person and the position. Important ones include name, unique personal identifier, ethnic, civil entrance examination rank, banner status, detailed geological information of the position, etc. The records cover a range of about three hundred years from 1644 to 1911. The distribution of number of records isn't even across years, with a focus on 1830 - 1912. However, there is no significant missing time period, thus it provides us a great resource to carry out spatio-temporal studies. The dataset was only recently made public in July 2019. [3]

4.2 Preprocessing Steps

We follow a similar way as [15] to process the data. There are a total of 2,896,206 appointment records in the latest Jinshenlu database and they need to be processed for arbitrariness and uncleanness. We follow the following steps to clean the data. Since we already included this part in our milestone report, we leave it out for supplementary materials.

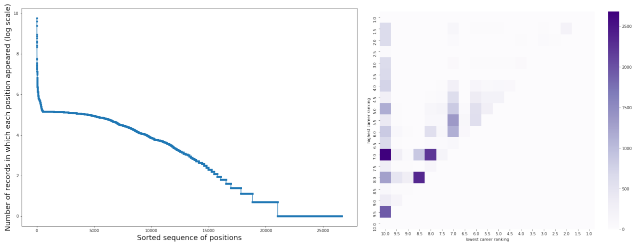
4.3 Data summary statistics and findings

We start exploring our dataset by visualizing a few high-level statistics we most care about.

A primary work is to figure out the exact definition of different entities that we examine, in the context of our actual dataset. The concept of "position" remains ambiguous. We explored several different ways of identifying a "position", and finally decided to use the most detailed granularity which combines all these identifiers including provincial level, county level, administration section, administration

[2]The dataset is available at `https://www.shss.ust.hk/lee-campbell-group/china-government-employee-database-qing-cged-q/`

[3]We would like to acknowledge that part of our project was built on the previous work done by our author Yanbang Wang. However, our work significantly extend those results. See section 8 for details.



(a) A somewhat mysterious exponential distribution of number of occurrences of positions: y-axis is the logged number of of occurrences of each position, x-axis is simply the sequence of positions indexed sorted by their y values.

(b) Correlation between starting postion's class and highest position's class in one's career. Stronger the correlation, darker the color.

Figure 2: Two visualizations of our dataset's feature distribution.

department, and detailed job title, which led to 26625 positions. A very interesting result is shown by Figure 2a, where Y axis denotes the logged number of of occurrences of each position, X axis is simply the sequence of position indexed sorted by their y values.

Another preliminary exploration relates to our investigation of highest class of position one can reach in his career. Specifically, we plot a heatmap where each cell denotes a (starting position's class, highest position's class) pair in people's career (so obviously only the upper half of the plot is valid). This serves as a condense, simplified summarization of people's career trajectories (by only focusing on two ends). Two interesting finds are revealed by this heatmap: First, it is slightly counter intuitive to see that starting at a lower-class position does not necessaries indicate a poorer chance of getting promoted to a high-class position later in one's career, as shown by the leftmost column of the heatmap. Second, it is noticed that the distribution of the colors is far from being continuous, or smooth. In particular, several "highlighted cells" are of high interest to us. We carried out further analysis of the darkest cell at (10.0, 7.0) by consulting domain experts, and was informed that exactly maps to a written rule of promotion: the government established a channel to regularly promote well-performed people from a wide range of lowest ranked positions.

As mentioned in the previous proposal of the project, constructing a career graph from our dataset yields a career network, in which each node is a position, and there exists a directed edge A →B if at least one person used to move from position A to B.

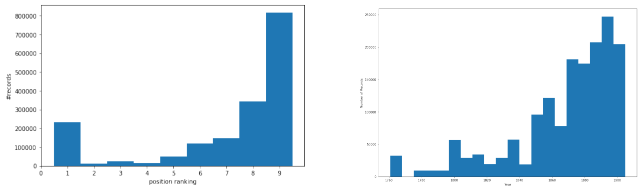
We also plot the distribution of position rankings and number of records in each year, so as for us to make more informed decision later when needed, as shown in Figure 3.

5. METHODOLOGIES

5.1 Network analysis and modeling

5.1.1 Evolution of properties

We investigate into the following properties in the networks, and see how they evolve as the network grows. The task is defined as in Section 3.1.



(a) Distribution of position rankings

(b) Number of records per year

Figure 3: Two more visualizations of our dataset's feature distribution.

Static exploration

- *In/Out Degree Distribution.* We plot the relationship between degree and number of nodes of that degree. Most real networks show scale-free properties.
- *Edge weight distribution* This property is examined here because the edges are weighted in our setting.

Dynamic exploration

- *Clustering coefficient* The clustering coefficient for a graph G is defined as $C = \frac{1}{N}\sum_i \frac{2e_i}{k_i(k_i-1)}$.
- *Reachability* Defined as the probability of path existing between two random nodes. Calculated through sampling.
- *Largest weakly connected component* Defined as the number of nodes in the largest connected component.
- *Diameter* Defined as the longest distance between any two nodes.

5.1.2 Modeling densification

To model the realistic network, we first find out the relationship between number of nodes and that of edges. In many realistic networks, it has been shown that their growth follow power law [8], where the number of nodes and that of edges satisfy: $E(t) = N(t)^{\alpha}$. On our dataset however, we claim that the two variables satisfy the exponential formulation, i.e.

$$N(t) = N - Ne^{-cE(t)/N}$$

Suppose the total number of positions in the records used to construct the network is N which is unknown to us. Another parameter c refers to the modified coefficient considering the effect of multi edges. We try to model the relationship of number of nodes and edges, so we calculate the expectation of increase in number of nodes, when we add a new edge at each time. Suppose currently there are $N(t)$ nodes in the network, thus $N - N(t)$ nodes haven't been added to the network.

We first note that considering the organization of our records in database, as we add a new edge, we always start from one existing node (previous position which has already been added to the network), and could either link it to previous node or introduce a new node. The probability of introducing a new node is proportional the the fraction of nodes yet to be discovered, times a modified coefficient c which takes into consideration the densifying edges in the graph, which

is a simplified assumption. The probability of introducing one new node is $p_1 = c * (N - N(t))/N$. The probability of introducing no new nodes is $p_0 = 1 - p_1$.
So each time we append a new edge in the network, the expected increase in number of nodes would be:

$$\frac{dN(t)}{dE(t)} = 1 * \frac{c(N - N(t)}{N} + 0 = \frac{c(N - N(t))}{N}$$

Solving the equation, given the starting point at N(0), E(0) = (0, 0), we have $N(t) = N - Ne^{-cE(t)/N}$ which proves our assumption. We'll try to fit our model to the realistic data in Section 6.1.

5.1.3 *Modeling real network*

Here we propose a method for modeling our career network, trying to fit the properties exhibited in realistic network. We conduct our algorithm in two stages, what we call as **Skeleton construction** and **Densification**. The first stage is adapted from forest fire model with orphans, introduced by Leskovec et al. [8]. Orphan is adopted here to avoid giant connect component that contains all nodes in the graph. After the first stage, we get a graph where every edge has weight 1. Starting from here, we then stop adding nodes, but increase edge weights instead. This stage is called Densification by us, where we increase edge weights in preferential attachment-like way, where stronger links get more weights. Given number of nodes in the graph, our full algorithm first estimates number of edges needed, then conducts the two-stage construction algorithm. Due to space limit, we put forest fire algorithm in our supplementary materials.

Algorithm 1 Two Stage Construction

```
Require: N, G                          ▷ Number of nodes, Graph G
  E = (N/c) log (N/(N−n))              ▷ Derived from
  for i = 0 → N do                     ▷ Add N nodes to graph
      ADDNODEFORESTFIRE(G, P, Q, O)
  end for
  e_R = E − G.edges                    ▷ Remaining edges to add
  DENSIFICATION(G, e_R)
```

Algorithm 2 Densification

```
Require: G, E                          ▷ Graph G, Number of edges E
  for iter = 0 → E do                  ▷ Add a new edge to graph
      ŵ_i = w_i/Σ_j w_j for all i.     ▷ Normalized prob.
      Sample a new edge e_S based on w.    ▷ Preferential Attachment
      Increase weight of e_S by 1.
  end for
```

5.2 Discovery of Stepping-stone Positions

The methodology for stepping-stone position discovery is adapted from our previous unpublished attempts [15]. See 8 for more details.
In 4.1 we described the method to construct networks for this task. According to the previous description of how people conceptualize the prospect of getting promoted from one position, we can formulate the calculation of a position's "potential" as the following:

$$potential_i = AGG(\{potential_j, j \in out\_neighbors(i)\})$$

where AGG can be some mannually designed permutaion invariant function such as mean, sum, or maxpool. There is likely "dangling node problem" with this formulation. See our supplementary materials for more details.[4]
We propose our method of calculating *potential* considering the aforementioned aspects.
We first noticed the following relationship satisfied for each node:

$$\text{value} = \text{nominal\_value} + \text{potential}$$

$$\Leftrightarrow \text{potential} = \text{value} - \text{nominal\_value}$$

It's implied in the first equation that the 'value' of a position could be seen as the summation of two parts: nominal values corresponding to the position and the potential that we defined above. We can either associate the given variable of *pin_ji* to nominal_value for a node/position, or we could utilize domain knowledge on the importance of the position's title. The nominal_value for each node is fixed to a known value in either case. The equation allows us to accomplish the goal by computing the "value" of a node instead although there could be problems when directly computing the potential of a node. This shift of focus solves the "dangling node" problem and also integrates more known information about the database into the whole picture.
Based on the above discussions and [15], we then give a recursive definition of a node's value:

$$\begin{aligned} value(u) = \sum_{v \in out(u)} & \alpha^{\bar{t}_{u,v}} \times p_{u,v} \times value(v) + \\ & \alpha^{\bar{t}_{u,u}} \times p_{u,u} \times value(u) + \\ & \alpha^{\bar{t}_{u,out}} \times p_{u,out} \times 0 + \\ & nominal\_value(u) \end{aligned} \tag{1}$$

In the above equations, v is u's out-neighbors, excluding u itself in case a self-loop exists. α is the time damping coefficient to attenuate the weight, set to 0.8 by default. $\bar{t}_{u,v}, \bar{t}_{u,u}, \bar{t}_{u,out}$ are the average time taken for one person in position u to move to another position (i.e position v, position u itself, or off record, respectively). $p_{u,v}, p_{u,u}, p_{u,out}$ are possibility for one person in position u to move to another position.
The equation can be rewritten elegantly into the matrix form:

$$Ax + c = x$$

Where $A_{u,v} = \alpha^{\bar{t}_{u,v}} \times p_{u,v}$, $A_{u,u} = \alpha^{\bar{t}_{u,u}} \times p_{u,u}$, $x_u = value(u)$, $c_u = nominal\_value(u)$. We can solve x using LU decomposition $I - A$, and potentials of all positions are $x - c$. An alternative is to use power iteration since the dimension (and thus memory needed) may be too high to some computing device[15].

[4] Found at https://docs.google.com/document/d/1xdEnYbobAfOsUzRnENpJoVdqDAAlHOtjEwQDksrmv4A/edit?usp=sharing

5.3 Prediction of Career Achievement

In this part we present methods to solve the three questions proposed in Section 3.3. We follow an incremental order of narrative. First, we introduce our baseline approaches: using only features of each individual without any relational information. Next, we report on the different types of relational information we selected, and how we integrate each of those relationships (as a graph) into our models. Finally, we also present how models built with different relationships are merged together to further improve prediction accuracy.

5.3.1 Prediction Using Individual's Features Only

Prediction under this scenario is a relatively standard supervised machine learning task: we train using features and labels from a portion of people, and test on the rest of them. The features are summarized in Appendix A. We trained using linear regression, random forest, or a shallow multilayer perceptron.

5.3.2 Iterative Regression Using Relational Information

Our iterative regression works as follows. We start by constructing different graphs based on some latent relationship of people in the database.

For example, it was mentioned in [6; 16] a very intersting phenomenon in China's history: there was a special group of people who almost by default consider themselves political allies – the small group of very top elites that excelled in the civil entrance test in the same year (called *Jiingshi*[5] *of the same year*). Since such alliance of people severely undermined emperor's sovereign power, it was strictly forbidden and hated by all the rulers and thus usually went under the table. We proceed by constructing a graph of latent allies based on this: each node in the graph represents a person, and there is an edge between two nodes if the corresponding two persons are *Jingshi of the* ***same*** *year*.

Other two graphs are constructed following similar procedures. One graph captures the similarity among positions: two nodes (persons) are connected if they served in the same type of position in their first appointment. The other graph captures the geographical relationship: two nodes (persons) are connected if they served in the same administration region (county level) in their first appointment.

We do training on each graph independently first, and on multiple graphs later. Note that in our setting, for each graph, we fix the same group of nodes as training set, the rest of them as test set. We do iterative regression by label propagation. In training phase, the inputs are the followings:

- a graph $G = (V, E)$
- an array of node indices I indicating which nodes in the G are training nodes in shape $(n_{train},)$, where n_{train} is the number of training nodes
- a feature matrix X_{train} in shape (n_{train}, d) where d is the dimension of each raw feature
- a feature matrix X_{test} in shape (n_{test}, d)
- an array, Y, of corresponding labels, in shape $(n_{train},)$
- N: number of iterations

The algorithm is shown below.

[5]Meaning most successful exam candidates

Algorithm 3 Iterative Training on One Graph

Require: $G = (V, E)$, I, X_{train}, X_{test}, Y, n_iter
1: $Regressor1 \leftarrow \text{FIT}(X_{train}, Y)$ ▷ Train a preliminary regressor using X_{train}, Y
2: $\hat{Y} \leftarrow \text{PREDICT}(Regressor1, X_{test})$ ▷ Use the regressor to intialize labels for test set
3: Initiate $\tilde{X}_{train}$ to an empty matrix with n_{train} rows
4: **for** $i = 0 \rightarrow N$ **do**
5: $node\_labels \leftarrow \text{COLLECTNEIGHBORLABELS}(i, G, \hat{Y})$ ▷ collect the labels of nodes neighboring to node i in G
6: $label\_features \leftarrow \text{COMPUTESTATS}(node\_labels)$ ▷ extract features from the node labels
7: Create a new feature for node i by CONCATENATE(X_{train_i}, $label\_features$)
8: Put the new feature to the i-th row of $\tilde{X}_{train}$
9: **end for**
10: $Regressor2 \leftarrow \text{FIT}(X_{train}, Y)$ ▷ Train a second regressor using $\tilde{X}_{train}$, Y
11: $\hat{Y} \leftarrow \text{PREDICT}(Regressor2, X_{test})$ ▷ Use the second regressor to predict the labels for test set

return prediction: $\hat{Y}$; model: $Regressor2$

Algorithm 4 COMPUTESTATS

Require: $node\_labels$: an array of node labels in real numbers

return mean, std, length, maximum, and minimum of $node\_labels$

The procedures above are similar to iteratie classification introduced in class, except that in each iteration, we aggregate information from the neighborhood by computing different statistics of the neighboring nodes' labels.

To train on multiple graphs G1, G2, G3, ...etc., we modify Line 5 of the main algorithm by replacing G with $(G1, G2, G3)$ and collecting labels based on all three graphs

5.4 Performance Metrics

Three metrics are used to measure regression's accuracy: R2 score, root of mean squared error, and classification accuracy. Note that to compute classification accuracy, we discretize prediction $\hat{Y}$ into the nearest 0.5 level and check if that matches with ground truth.

One more thing to notice is that, since a graph may only connects a very small percentage of nodes (people), we should only expect accuracy to be (hopefully) increased on nodes (people) that are connected by the graph (or nodes that have at least one edge of the graph). Therefore, to be able to compare effectiveness of different graphs, we also compute a second group of metrics only on those people connected by the graph (or *affected people* in our terminology)

6. EXPERIMENTAL RESULTS

6.1 Results on Task 1: Network Analysis and Modeling

6.1.1 Evolution analysis

We first give a glance into the degree distribution as well as weight distribution in the real network. For degree distribution, we plotted both out-degree distribution and in-degree

distribution of nodes. Not surprisingly, the results exhibit heavy tail property on both in/out degrees, and are shown as power law. Most positions only have very few out links or in links, with some positions playing the role of 'hubs'.
Different from many real networks that have been studied, the edges in network we constructed are weighted, so we also look at the distribution of edge weights.

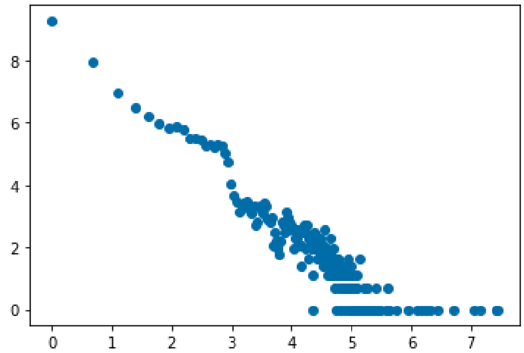
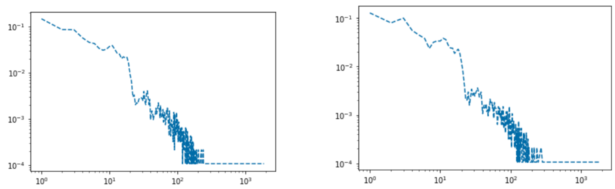


Figure 4: Weight distribution of edges in real network.

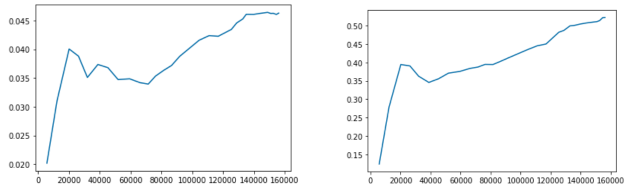
As we can see in Fig 4, similar to degree distribution, the weights also seem to obey power law, exhibiting heavy tail on the right part of the curve.
The observations go well as expected actually. In Section 6.2, we will see the distribution of potential of different positions are quite imbalanced. The heavy-tailed in/out degree as well as heavy-tailed weight distribution could be used to give some hints on the empirical experimental results. Most nodes in the network are linked with only very few neighbors, which means that people who have been placed in the position are not believed to have a large potential to be promoted to other higher ranked positions.
We then examine other properties and see how they evolve as the network grows. We plotted properties including clustering coefficient, size of largest connected component, reachability as well as diameter in figures respectively.



(a) Out degree distribution (b) In degree distribution

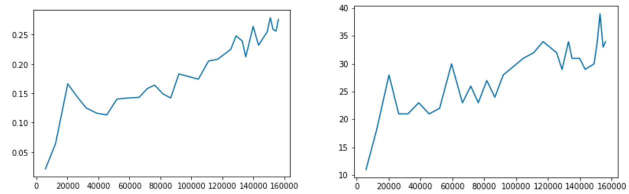
Figure 5: Plots on out/in degree distribution on real networks.

Generally, as can be observed from the plots, reachability, size of largest weakly connected component as well as clustering coefficient and diameter exhibit ascending trends. At the beginning of the evolution, the edges tend to be sparse, with many separated components, thus low in these metrics. As more edges come, components begin to be connected, forming large connected ones. The reachability of nodes increases as well. Also, previous research indicates that a large number of promotions take place in the same district and the same levels, leading to local structures in



(a) Evolution on clustering coefficient in the network. (b) Evolution on size of largest weakly connected component in the network.

Figure 6: Two plots on evolution.

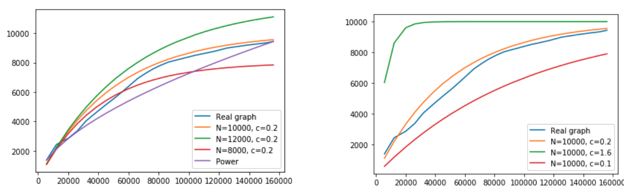


(a) Evolution of reachability (b) Diameter evolution

Figure 7: Plots on reachability and diameter.

the networks. As more edges appear in the network, the local structures are revealed and this leads to the increasing clustering coefficient.
The behavior on evolution of diameters looks weird at first sight. Many real networks are thought be shrinking in diameter in spite of densification in edges [8]. An increasing trend is observed in our poster. However, the organization of the records might give us a hint. Since we process the records sequentially, the records that belong to the same person naturally forms a path in the graph, the path gets longer as more records from the same person arrive. In this way, as the network grows, the path is also expected to be on an ascending trend, and saturates as fewer new edges emerge later.
These patterns actually reveal the trend that there had been rather frequent promotions across different positions. The range of promotions cover a vast majority of the network, while maintaining high density in local neighborhood. The properties resemble those in social networks which we usually see in reality.

6.1.2 *Densification Modeling*



(a) Effect of parameter N (b) Effect of parameter c

Figure 8: Performance on different settings of parameters.

In this part, we try out different settings of parameters in our exponential model, and see how it fits the realistic distribution.
In Fig 8, the two plots show the effect of parameter N and

c on the shape of our generated curve. The blue curves denote the realistic growth curve. We can see that when we set $N = 10000, c = 0.2$, our curve fits the real data fairly well. Also note that we're comparing our curve with that generated by fitting with power law densification where $E(t) \propto N(t)^{\alpha}$[8]. The plot shows that power law doesn't apply well here, showing a relatively too flat curve.

6.1.3 *General Properties Modeling*

In this part we continue with our verification of our model on other important properties. To compare with, we implemented two simple baselines as described below. For all the models as described including our model, we fix the number of nodes to 9000 and edges to 15000, and generate corresponding graphs.

1. **Null Model** Given number of edges, each time two sides of an edge are randomly selected. Repeat until all edges have been placed in the graph.

2. **Preferential Attachment** For this method, we start with en empty graph of given number of nodes, then we repeat adding edges to the network, sampling based on the weights of each edge. The implementation differs from the original paper in that in that paper each node has fixed number of out links, which doesn't apply here.

We empirically evaluate the generated graphs generated from different methods. The results in the following table list the empirical results, followed by our analysis. To make it clear, Based on the analysis in previous sections, our goal is to generate the graph that produces large clustering coefficient, medium connected component size as well as reachability, large diameter, heavy-tailed in/out degree distribution as well as weight distribution.

Properties	Null	Pref Att	Ours
Clustering Coeff	✗	✗	✓
Reachability	✗	✗	✓
Diameter	✓	✓	✗
WCC size	✗	✗	✓
Degree distr	✗	✓	✓
Weight distr	✗	✗	✓

We can conclude from the above table that our method performs the best on the metrics, except that it doesn't do well in diameter.

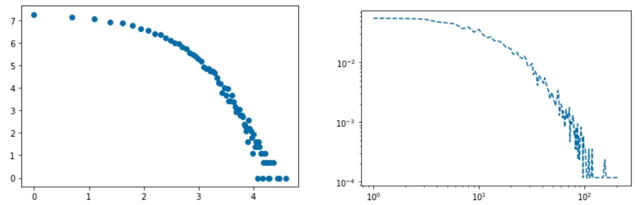
The randomness in the other two methods leads them to disregard local structures in the network, thus they tend to connect different parts in the graph together and producing huge size of connected component, as well as a strong reachability after certain iterations. This would also result in a low clustering coefficient.

The preferential attachment method naturally prefers to attach edges to those high degree nodes, leading to heavy-tailed distributions. However, since the network we're simulating is relatively sparse compared to complete graph, the weight distribution is rather sparse, degraded to several points instead of power law.

Our model produces a high clustering coefficient of 0.132, reachability of 0.17, largest WCC of 0.37, and diameter of 6. Actually, our method models the properties through the two stages. For the forest fire stage, the skeleton of graph is established, considering the local structures. The orphan used in the method prohibits the model from forming a huge component containing all the nodes, as well as limiting reachability to reasonable scale. The second stage takes the distribution into consideration. The preferential attachment-like process endows the weight distribution with heavy tail. This also pushes the tail of in/out degree distribution further, strengthening the power law.

The limitation of our generated graph lies in the diameter, where we generated small diameter, compared to that of real ones. We think this reflects the property of shrinking diameter that lies in forest fire model. Not taking the trajectories of individuals when constructing graphs might lead to the problem, but more rigorous derivation needs further elaboration.

For reference, Fig 9 shows the out-degree and weight distribution of our generated graph.



(a) Weight distribution on generated graph.

(b) Out-degree distribution on generated graph.

Figure 9: Distribution on generated graph.

6.2 Results for Task 2: Stepping-stone Positions Discovery

As we mentioned above, the results on SSP discovery mainly come from previous attempts (see section 8 for details.)

6.2.1 *Preliminary analysis*

After calculating the potential of all positions as described in Section 5.2, we first give a glance over the statistics on the distribution of potential. See 10.

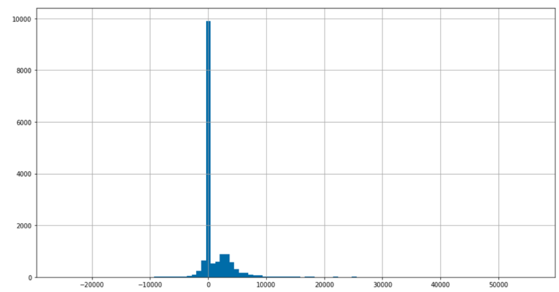


Figure 10: Distribution of potential over all positions, where y-axis is the number of records and x-axis is potential.

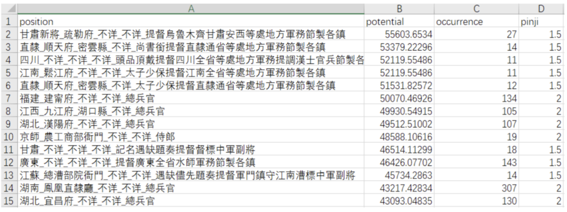
At the same time, we also listed the highest ranked positions in 11.

We have some observations from the preliminary statistics as well as other results.

- Many highly ranked positions are related to military service to some extent, despite the fact that in the data pre-processing stage we've already eliminated all the military positions from the dataset. We consider this as indication that warfare plays an important role in a person's career development.

- Interestingly, although the positions that belong to higher nominal valued classes have a higher chance to be ranked higher, it goes to the other side when we calculate the normalized version of potential, which is obtained if we divide the potential value of a position by its nominal value. A possible explanation is that for such positions, people have less room for further promotion to even higher positions.

- We consider both positions with the largest and smallest (negative value) potentials to be highly ranked ones.



	A	B	C	D
1	position	potential	occurrence	pinji
2	甘肅新將_疏勒府_不详_不详_提督烏魯木齊甘肅安西等處地方軍務節製各鎮	55603.6534	27	1.5
3	直隸_順天府_密雲縣_不详_尚書銜提督直隸通省等處地方軍務節製各鎮	53379.22296	14	1.5
4	四川_不详_不详_不详_頭品頂戴提督四川全省等處地方軍務提調漢士官兵節製各	52119.55486	11	1.5
5	江南_鬆江府_不详_不详_太子少保提督江南全省等處地方軍務節製各鎮	52119.55486	11	1.5
6	直隸_順天府_密雲縣_不详_太子少保提督直隸通省等處地方軍務節製各鎮	51531.82572	12	1.5
7	福建_建甯府_不详_不详_總兵官	50070.46926	134	2
8	江西_九江府_湖口縣_不详_總兵官	49930.54915	105	2
9	湖北_漢陽府_不详_不详_總兵官	49512.51002	107	2
10	京師_農工商部衙門_不详_不详_侍郎	48588.10616	19	2
11	甘肅_不详_不详_不详_記名遇缺題奏提督督標中軍副將	46514.11299	18	1.5
12	廣東_不详_不详_不详_提督廣東全省水師軍務節製各鎮	46426.07702	143	1.5
13	江蘇_總漕部院衙門_不详_不详_遇缺儘先題奏提督軍門鎮守江南漕標中軍副將	45734.2863	14	1.5
14	湖南_鳳凰直隸廳_不详_不详_總兵官	43217.42834	307	2
15	湖北_宜昌府_不详_不详_總兵官	43093.04835	130	2

Figure 11: Some examples of highest ranked positions as well as their statistics.

6.2.2 *Examining Subgraph using BFS*

To verify the effectiveness of our proposed method, we look into several highest ranked positions in the network. We especially pay attention to the patterns in the adjacent nodes of our target: do they lead to highly ranked nodes as we expect? To reach our conclusion, we do a breadth first search on the graph (See 12), starting from the node circled in red. This position is ranked high by our algorithm.

Considering the fact that the links are weighted and directed, we can see from the graph that obviously people have a high chance starting from our target node to go to highly ranked position clusters at bottom right of the graph.

Figure 12: A Subgraph Extracted with BFS.

6.2.3 *Robustness Analysis*

Since this part comes from previous attempts (see section 8), we refer you to the shared doc mentioned above due to space limit.[6]

6.3 Results on Career Path Prediction

We implemented three baseline models that trained on each person's features seperately (without using any graph information). The regressors used as baselines are linear regression, a single layer neural network (MLP, with hidden size = 100), and random forest (number of trees = 100). Recall that a higher accuracy and R2 score indicate better performance, and mean error the opposite. As shown in Table 1, the three models achieve similar performance and are all unsatisfactorily below 35% accuracy.

Since random forest is the best baseline, we stick to it in our iterative methods. In iterative methods, three graphs are used: "position graph" where people (nodes) are connected if they have the same position, "*jingshi* graph" where people (nodes) are connected if they both excelled in civil test of the same year, and region graph where people (nodes) are connected if they used to serve in the same place in the same year. We also report the performance of a model that used all three graphs. As shown in Table 1, they all achieve different levels of improvement over the baselines. Note that the model trained on *jingshi* graph do not lead to a large boost in overall accuracy when tested on the complete test set. That is because the connected components of *jingshi* graph only account for less than 9% of the whole graph. In other words, we should only be able to observe more accurate predictions on a very small number of people. However, it turned out that when we tested performance on people that are connected by *jingshi* graph, that is, a subset of the complete test set, a significant improvement in accuracy was observed, which we will elaborated next.

	Classif.	Graph	Acc.	R2 Score	Mean Error
		Baselines			
1	Linear	none	0.323	0.212	1.76
2	MLP	none	0.333	0.238	1.73
3	RF.	none	0.347	0.246	1.71
		Iterative Methods			
4	RF.	position	0.680	0.663	1.14
5	RF.	*Jingshi*	0.357	0.258	1.70
6	RF.	region	0.418	0.482	1.42
7	**RF.**	**combined**	**0.722**	**0.712**	**1.05**

Table 1: Performance on complete test set

Figure 13 shows how iterative regression achieves a much better performance than baselines over iterations. Lines of different colors represent performance of models trained on different graphs. Among the two models trained on each graph, the solid line corresponds to the model tested on complete test set, and the dashed line corresponds to the model tested on "the subset of test nodes that have at least

[6]This part could be found at `https://docs.google.com/document/d/1xdEnYbobAf0sUzRnENpJoVdqDAAlHOtjEwQDksrmv4A/edit?usp=sharing`

one edge in the corresponding graph". Therefore, the five solid lines, corresponding to Model 3 through 7 in Table 1), all have the same starting position at iteration 0, since initialized using the model represented by black line. In contrast, the dashed lines have different starting positions, because they are tested on a different test subset of nodes, even though they are also initialized using the model represented by black line.

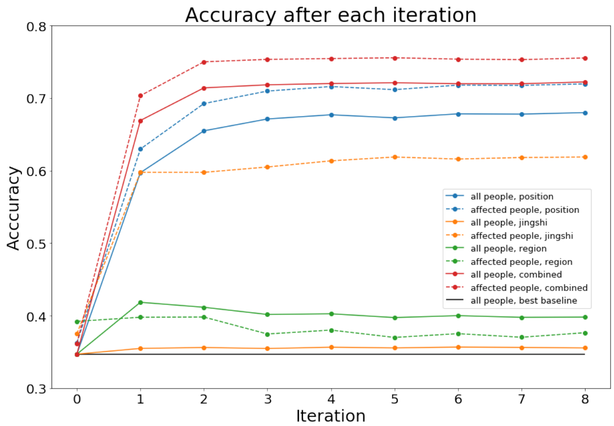


Figure 13: Comparison of different model's accuracy on complete/"affected" test set over each iteration. "Affected" means subset of nodes in the test set that have at least one edge in the corresponding graph.

It can be observed that combining all three graph consistently yields better performance than using single graphs. All models roughly converge after three iterations.

It is in some sense amazing that more than 0.7 accuracy can be achieved using only people's background and first appointment's information. **It is strongly evident that as far as our dataset indicate, people's career was determined to a great extent by the time they enter the feudal system.** Given the (arguably) decent accuracy of 0.7+ in prediction career achievement, we would also like to point out that this would hardly be possible without leveraging the latent relationships between people. As we observed with baseline models, the amount of clues about future career provided a single person can be very limited (acc. is only 0.3+). **In that sense, it is fairly reasonable to say that "trends" exhibited by large group of people collectively have very strong implications about their career.**

7. DISCUSSIONS

In this project, we explored the CGED-Q dataset recently released that contains China Government Employment data in Qing Dynasty. We are the first to crack the problems from the perspective of networks on this dataset. We thus conclude in this section the interesting patterns we've discovered, as well as exciting findings which give new insights into the classical topics, and which provide hints on promising directions of research.

From the static and flat records, we first construct graphs where nodes represent positions and edges represent promotions. We dive into analyzing attributes of our network, including static and dynamic properties. We discovered that the properties resemble those in social networks to some extent. These properties interestingly reveal some patterns and existing conclusions about promotion systems in the history. We also proposed a model that models the properties well, compared to other random models.

The heavy-tailed node degree distribution as well as weight distribution indicates that not all nodes are equivalent with regard to promotions. This gives rise to our research into discovering stepping stone positions. Aggregating information from neighbors, as well as utilizing the nominal values resulting from previous investigation, our method generates reasonable rankings, verified by our case study.

Stepping from macro-view to micro-view, how will a person perform in his future career? How can we predict his career achievement? Predicting from his attributes is one way, but we're inspired by specific domain knowledge and construct graphs to exploit the information to the maximum. Iterative classification method is illustrated in different scenarios, and our combination of heterogeneous sources information demonstrates superior performance on the task.

One promising extension in the future is what conclusion can we reach regarding social mobility? Some research work showed that most promotions occur in the same region or across similar levels. In other words, are there communities/clusters in the position network? Are there interesting patterns in the career path of a specific person?

Another question we'd like to ask is, since there are significant events taking place in history, known to transform the structures of China society to a large extent, can we observe patterns in around the occurrence of such events? This requires us to conduct more in-depth temporal analysis on the network.

8. ACKNOWLEDGEMENTS

There are several things we'd like to make some clarifications/acknowledgements on. They're listed below.

- **We consider the contributions of the two authors to be equal.**
- We acknowledge Lee-Campbell Group for their endeavors in building the Jinshenlu database, which provides us with precious sources of information when we build our models and conduct experiments. Due to data protection contracts, Jingjing works on the publicly available dataset, while Yanbang works on full private data.
- As we clearly mentioned in our milestone, part of our work is adapted from the unpublished attempts of one of our author Yanbang Wang, namely the efforts on stepping stone discovery. This includes the problem definition, method and results on stepping stone discovery. All remaining contributions are made for this class (cs224w).
- We recommend you read our supplementary materials[7]. Although this doesn't affect the expression of our major idea, but helps you understand our work better.

[7]`https://docs.google.com/document/d/1xdEnYbobAf0sUzRnENpJoVdqDAAlHOtjEwQDksrmv4A/edit?usp=sharing`

9. REFERENCES

[1] F. Amblard, A. Casteigts, P. Flocchini, and W. Quattrociocchi. On the temporal analysis of scientific network evolution. pages 169–174, 10 2011.

[2] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.*, 7:2399–2434, Dec. 2006.

[3] C. D. Campbell, B. Chen, Y. Ren, and J. Lee. China government employee database-qing (cged-q) jinshenlu 1900-1912 public release, 2019.

[4] L. C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35, mar 1977.

[5] S. Grayson and D. Greene. Temporal analysis of reddit networks via role embeddings, 2019.

[6] T. Ke-jun. Ways of promotion in ming dynasty. *Journal of Historical Science*, 1, 2004.

[7] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, Sept. 1999.

[8] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graph evolution: Densification and shrinking diameters, 2006.

[9] G. Li, T. Zhou, and H. Xu. *The Quantitative Analysis of Personnel Transition in Qing Local Governance*. National science council, 1975.

[10] J. Neville and D. Jensen. Iterative classification in relational data. In *Proc. AAAI-2000 Workshop on Learning Statistical Models from Relational Data*, pages 13–20, 2000.

[11] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

[12] U. Nieminen. On the centrality in a directed graph. *Social Science Research*, 2(4):371 – 378, 1973.

[13] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.

[14] Y. Sun and J. Han. *Ranking Methods for Networks*, pages 1488–1497. Springer New York, New York, NY, 2014.

[15] Y. Wang. Analyzing career trajectories of qing dynasty civil servants with data mining and machine learning, 2019.

[16] A. Wedeman. *Double paradox: Rapid growth and rising corruption in China*. Cornell University Press, 2012.

[17] R. Yuxue, B. Chen, X. Hao, C. Campbell, and J. Lee. China government employee dataset-qing dynasty jinshenlu 1900-1912 public release user guide. 2019.

[18] X. Zhu and Z. ghahramaniH. Learning from labeled and unlabeled data with label propagation. 2002.

APPENDIX

A. FEATURES USED BY BASELINE MODELS IN TASK 3

Feature	Explanation
year	year of the first position's appointment
season	season of the first position's appointment
exam	whether the person entered by taking the civil exam
qiren	whether the person is a member of royalty
mongol_banner	whether the person is a Mongolian banner man
han_banner	whether the person is a Han banner man
jigou_type	what type of admin. department was he serving
dynasty	dynasty of the person's first appointment