Analysis of E-taxi Market and Policies in Different Cities

Fan Jinlin

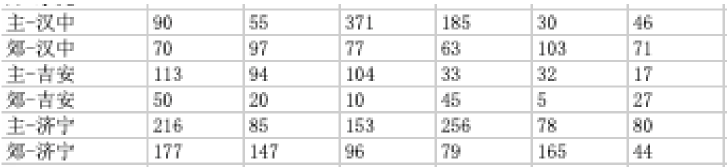
2019-Spring: AAS Final Report

1 Introduction

E-taxies are more and more important in people's daily choices of transportations. Not only for the convenience but also for the cheaper price and much better customer experience. Recently, lots of events come in front of public attention. Some of them are good, while some of them are bad. Taking the safety of passengers into consideration, government and relevant companies publish some items and lists to control the quality of the E-taxies and drivers.

All the analysis in this project will focus on some senario under this background based on true data collected by students in Law School of RUC. These datasets contain information about the requirments to taxis in several cities, news about certain events happened in several cities and competition between different taxi companies.

To get a deeper insight about the balance or the relationship between government's supervision strength and lobbying power of market and commercial cooperations, I will try to construct a frame for attaining quantified strength of supervision and monopolization of taxi companies, which can be used for further clustering of cities in order to find out similar patterns about cities in one same cluster. Some of the results will make sense, and others may be surprising, but useful for leading us to a brand new branch in the logical chain.



主-汉中	90	55	371	185	30	46
郊-汉中	70	97	77	63	103	71
主-吉安	113	94	104	33	32	17
郊-吉安	50	20	10	45	5	27
主-济宁	216	85	153	256	78	80
郊-济宁	177	147	96	79	165	44

Figure 1: Some samples in dataset1

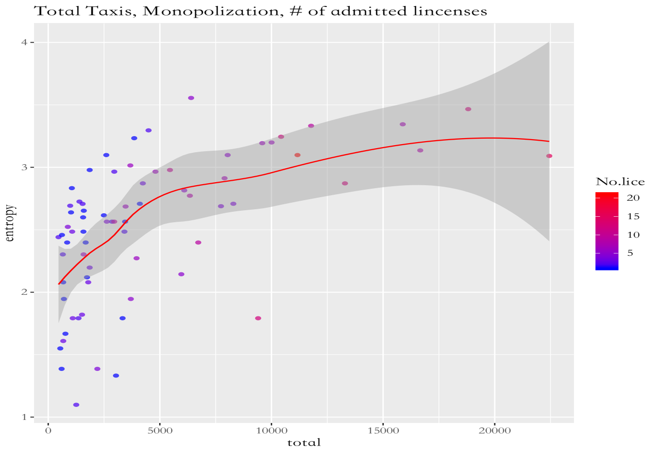


Figure 2: This figure shows that larger amount of taxies in a city means smaller monopolization power(larger entropy), and more companies can reduce the monopolization also.

2 EDA

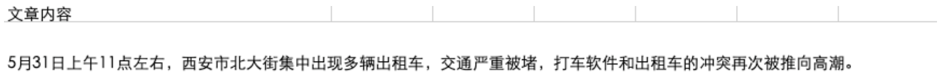
In this part, I will show what my datasets look like, and breifly express some results after exploratory data analysis. Basic tools for data processing also will be referred if necessary.

One excel table includes the total number of taxis for different companies, and the companies are divided into two groups, downtown or not. Some samples are in Figure 1.

Motivated by the entropy, which is a core concept in information theory, and defined as

$$l = -\sum_i p_i log p_i.$$

Entropy is used to measure unstability in a dynamic system, and in statistical cases, if n observations come from a uniform distribution, then the entropy



文章内容

5月31日上午11点左右，西安市北大街集中出现多辆出租车，交通严重被堵，打车软件和出租车的冲突再次被推向高潮。

Figure 3: One news in dataset

of these n obs attain the maximum. On contrary, if the entropy is smaller, the latent distribution is further from uniform, which means the system has kind of information about certainty, such a thing can also be explained as monopalization power in an unfair market.

In Figure 2, there are three features, every point stands for a city. And the color means the number of permitted taxi companies in this city. The fitted curve

Then I can also simply show datasets containing news in Figure 3. Except the content of news, the datasets also contain cities appearing in this news. Text-mining methods for these texts will be discussed in next section.

I use a map to express basic numeric features about different cities in Figure 4. This graph capture some advantages of using map and it contain many other information. Usually, such a map-type figure, sometimes, may express many other interesting patterns exsiting in geological aspects. But in my case, because of lack of data for other cities, there isn't any significant pattern we can get only from this graph.

3 Text-mining and LDA

In last section, we have briefly checked datasets about news. In this section, I will use some basic NLP (natrue language process) methods to handle these chinese-characteristics news, and my aim is to mine some useful message in news content, which can help me get the emotion index of different events. Such emotion index in one city may casue big effects on the behaviour of government.

Firstly, words segmentation is necessary for long text. There are many methods for segmentation, some are based on rules, such as Maximum Matching, Inverse Maximum Matching, while others are based on statistics, such as HMM (hidden markov model), N-gram method, CRF (conditional random fields). And I choose HMM for words segmentation.

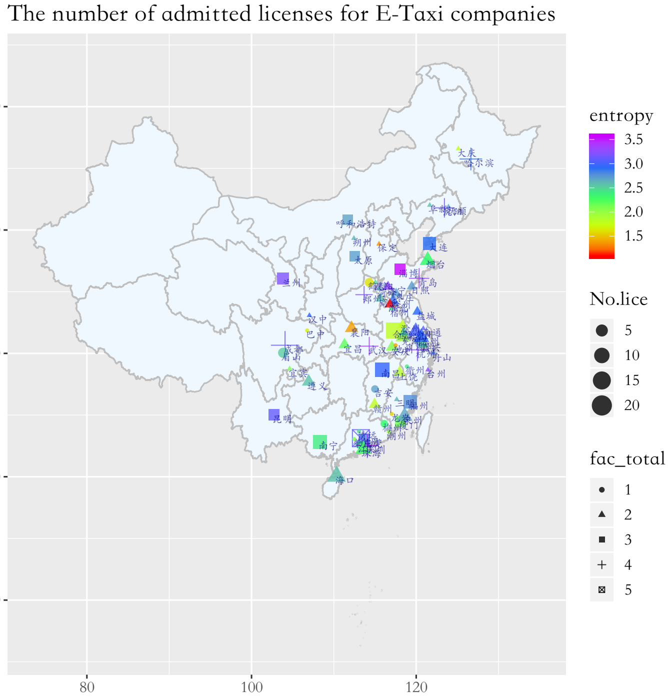


Figure 4: This graph contain more geological information. Although not significant, one interesting thing is that, severe monopolaztion cases usually are not big cities. Most of them are second-level or lower-level cities.

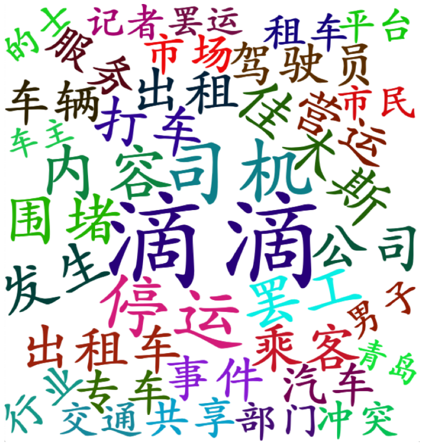


Figure 5: Top key words in all the news texts, it's easy to see that DiDi is the most mentioned company, and many words guide us to some accident.

3.1 Word segmentation

For this model, obviously, behind what we observe, there is a latent markov chain control the generation pattern of observations. In HMM, we assume Q is the set of states, V is set of possible observations, and N is the number of states, M is number of observations.

For a sequence with length T, I is respective state sequence, O is observations sequence. There is two important assumption, one is the markov property,

$$a_{ij} = P(i_{t+1} = q_j \mid i_t = q_i)$$

thus, we have a markov chain. The other is assumption about independent observations, which means

$$b_j(k) = P(o_t = v_k \mid i_k = q_j).$$

Here, the words in dictionary is set of observations and different combination of different words express different hidden states.

What's more, because these news are published on wechat platform, there are many useless words, such as ’ 阅读原文’, ’ 点赞’, ’ 关注’, ‘点击上方蓝字’ and so on. And usually quantifiers do not contribute any to our further sentiment analysis, so I filter them out. A visual representation for words after being segmented is represented in Figure 5.

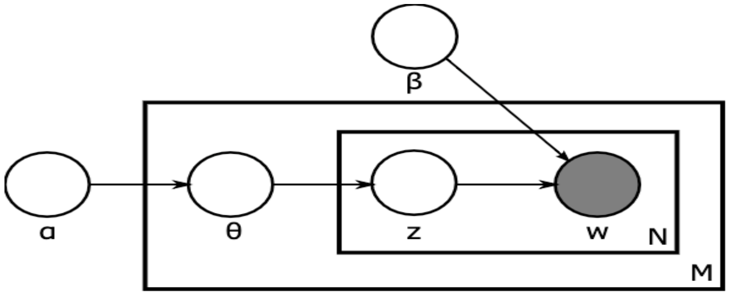


Figure 6: A draft of LDA model

3.2 LDA Topic Model

In this subsection, I will find out whether there are some seperable topics in all the news. LDA (latent dirichlet allocation) is a model in bayesian statistics. It is based on dirichlet distribution, which is a well-known conjugate prior for multinomial distribution. Multinomia distribution is also a good approximation for many multi-levels classsification tasks in real life.

In natural language processing, latent Dirichlet allocation (LDA) is a generative statistical model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar. For example, if observations are words collected into documents, it posits that each document is a mixture of a small number of topics and that each word's presence is attributable to one of the document's topics. LDA is an example of a topic model.

In LDA, each document may be viewed as a mixture of various topics where each document is considered to have a set of topics that are assigned to it via LDA. Each document is assumed to be characterized by a particular set of topics. This is similar to the standard bag of words model assumption, and makes the individual words exchangeable.

In Figure 6, the boxes are ”plates” representing replicates, which are repeated entities. The outer plate represents documents, while the inner plate represents the repeated word positions in a given document, each of which position is associated with a choice of topic and word. M denotes the number of documents, N the number of words in a document. The variable names are defined as follows: $\alpha$is the parameter of the Dirichlet prior on the per-document topic distributions, β is the parameter of the Dirichlet prior on the per-topic word distribution, θ_i is the topic distribution for document

i, φ_k is the word distribution for topic k, z_{ij} is the topic for the j-th word in document i, and w_{ij} is the specific word.

And the generative process is, if we have a corpus D consisting of M documents each of length N_i:

- Choose $\theta_i \sim Dir(\alpha)$, where $i \in \{1, ..., M\}$ and $Dir(\alpha)$ is a dirichlet distribution with a symmetric parameter α which typically is sparse ($\alpha < 1$)
- Choose $\phi_k \sim Dir(\beta)$, where $k \in \{1, ..., K\}$ and β typically is sparse.
- For each of the word positions i, j, where $i \in \{1, ..., M\}$, and $j \in \{1, ..., N_i\}$
  - Choose a topic $z_{i,j} \sim Multi(\theta_i)$
  - Choosse a word $w_{i,j} \sim Multi(\phi_{i,j})$

With the generative process above, we can do the bayesian sampling based on Gibbs sampler. The exact sampling condtional distribution is complicated, while there is a faster smapling method,

$$p\left(Z_{d,n} = k\right) \propto \frac{\alpha\beta}{C_k^{-n} + V\beta} + \frac{C_k^d \beta}{C_k^{-n} + V\beta} + \frac{C_k^w \left(\alpha + C_k^d\right)}{C_k^{+n} + V\beta}$$

In the Gibbs sampling, for convenience and speed, I choose 5000 interations without 1000 burn-in iterations, and set 5 topics in total. The topic assignments are in following table.

topic index	1	2	3	4	5
news count	134	242	222	696	121

Based on the words segmentation, I also give a estimated score for the sentiment of these topics. The sentiment of a news can be judged by counting the positive words and negtive words, this thing can be easily done according to the equation

$$S_i = \sum_j 1\{W_{ij} \in \mathcal{P}\} - 1\{W_ij \in \mathcal{N}\}, i = 1, ..., N, j = 1, ..., M_i,$$

where S_i is the score for i-th news, positive sign means positive sentiment, negtive cases are similar, and larger absolute value means stronger emotion.

$$S_{T_i} = \lambda m(S_j) + (1 - \lambda)\tilde{m}(S_j), i = 1, ..., 5, j \in \{1, .., N : S_j \in T_i\},$$

where $m(\cdot)$ is mean function, and $\tilde{m}(\cdot)$ is median function. From the score of different topics, we can know which news are positive, and for the negative topics, we can figure out those factors causing negative sentiment.

The table shows score we got for these 5 topics. Obviously, 2 and 4 are positive, well others are not.

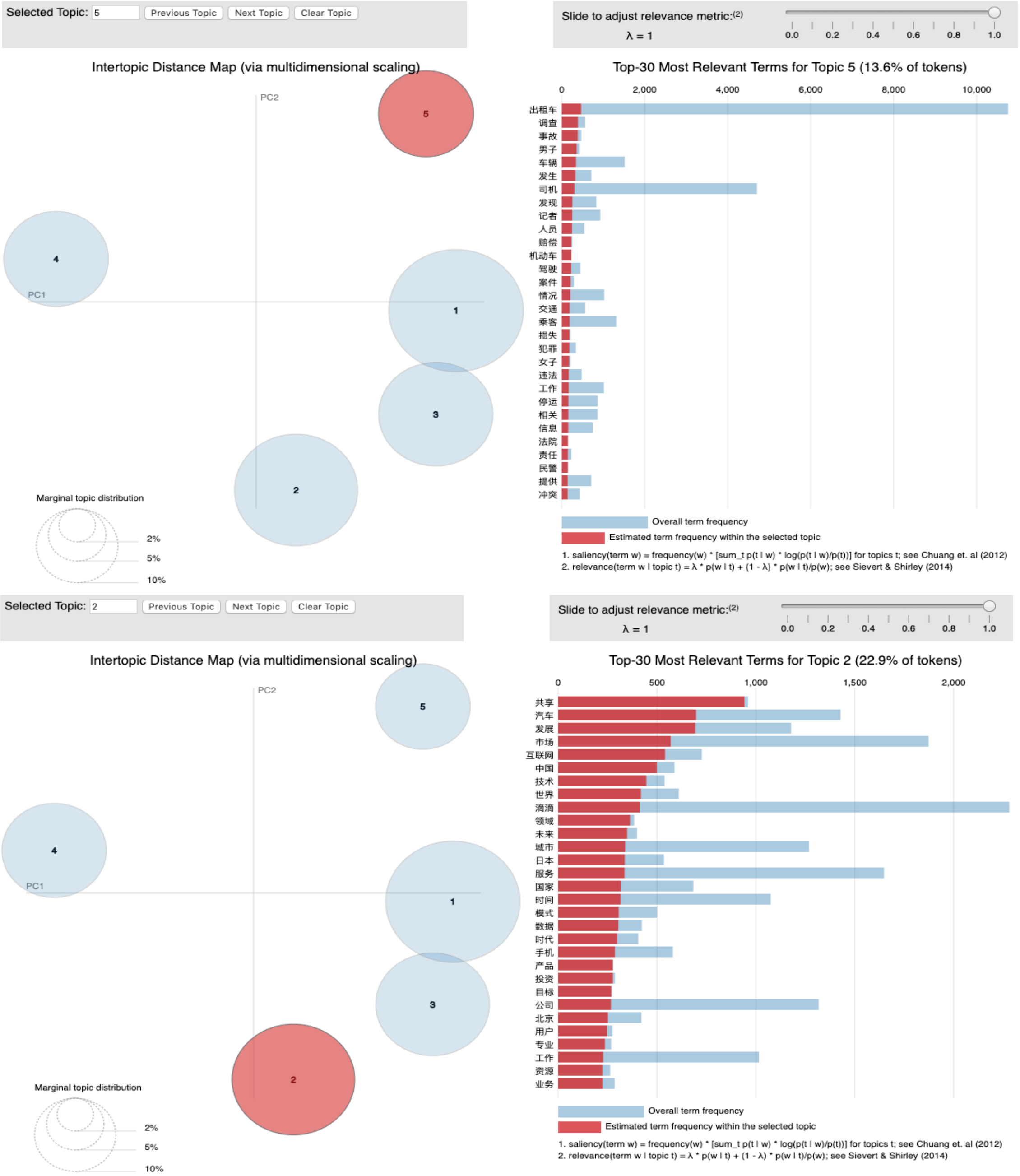
topic index	1	2	3	4	5
mean score	-2.4	5.86	-9.33	2.41	-3.77
median score	0.5	2	-7	1	-1

In figure 7, we can also check that the topic distribution, and key words in different topics. The sentiment divergence is quite obvious. In topic 5, many key words indicate that these articles are invovled with some accidents. While, in topic 2, these articles try to discuss relationship between rapid development of society and development of E-taxies market.

These two examples indicate that my procedure is reasonable in some way.

4 Clustering

After finishing last part, I can achieve my final goal. I want to see what kind of effects the news can produce on cities' policies. For doing this, converting some policies (necessary requirments to taxies actually) into numerical form is necessary, such as least driving distance, structure of engine, price of cars, least driving time, number of insurances and so on. After scaleing all the feature with maximum 5, I give a rating to measure intensity of policy with maximum, because this requirments provided by government are hard-thresholding, if one want to be permitted, he must satisfy all conditions. Choosing the maximum as rating R_i can preserve this property in requirments. Besides, after some experiments, I find that a simple weighted average is also useful. Both rating methods can properly measure intensity of policy in a city. Here, the i-th component means the i-th city. The rating equation



Selected Topic: 5

Slide to adjust relevance metric:(2)

λ = 1

Intertopic Distance Map (via multidimensional scaling)

Top-30 Most Relevant Terms for Topic 5 (13.6% of tokens)

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Selected Topic: 2

Slide to adjust relevance metric:(2)

λ = 1

Intertopic Distance Map (via multidimensional scaling)

Top-30 Most Relevant Terms for Topic 2 (22.9% of tokens)

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 7: This figure shows the difference between positive topic 2 and negtive topic 5. It is actually a screen shot of a shiny page which is supported by LDAvis.

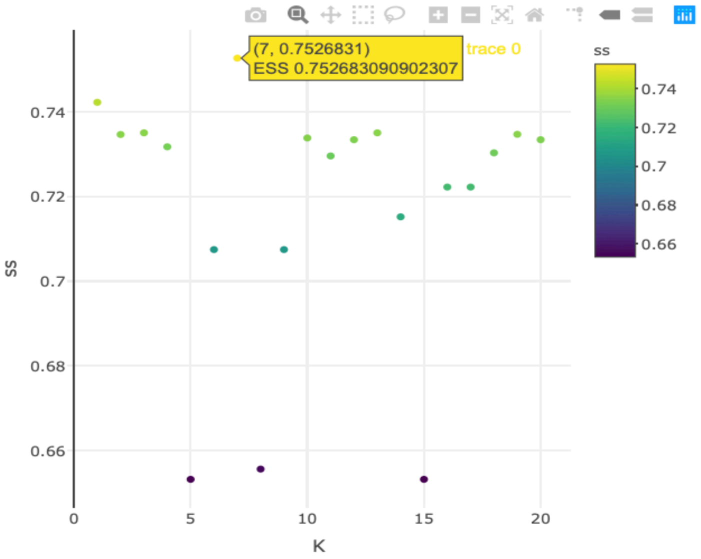


Figure 8: Find the best number of clustering according to explained sum of square.

$$\tilde{X}_{ij} = X_{ij}/max_i(X_{ij}) * 5, i = 1, ..., n, j = 1, ..., p$$
$$R_i^{(1)} = max_j(\tilde{X}_{ij}), R_i^{(2)} = \sum_j w_j \tilde{X}_{ij}$$

is easy to calculate, and we can find that such a combined feature benefits a lot to clustering procedure. Because size of sample is small, too many feature will cause curse of dimensionality, which will make distances between points larger resulting in bad clustering performences.

For clustering, three methods, DBSCAN, meanshift and k-means, have been tried. Not surprisingly, k-means gives out the most reasonable result, the reason is that sample sizie plays still an important role in clustering under such a dimension with $p \approx \frac{n}{5}$, where n is 71.

To set the number of clusters K in k-means methods, I check the plot of explained sum of square in Figure 8, which shows that $K = 7$ is a good choice.

And the cluster table is like this,

cluster index	1	2	3	4	5	6	7
count of cities	23	5	16	12	8	2	5

Here are more details, now we can discuss several clusters

- Cluster 1: ” 三明” , ” 上饶”, ” 商丘” , ” 嘉兴” , ” 宜宾”, ” 宿州” ” 抚顺” ” 朔州” , ” 梅州” , ” 江门” , ” 泉州”, ” 海口” , ” 清远”, ” 濮阳” ” 烟台” , ” 珠海” , ” 眉山” , ” 舟山” , ” 菏泽” , ” 衢州” , ” 遵义” , ” 阜新” , ” 马鞍山”

- Cluster 2: ” 南宁” , ” 厦门” , ” 哈尔滨”, ” 沈阳” , ” 福州”

- Cluster 3: ” 保定”, ” 合肥”, ” 大庆”, ” 安庆”, ” 宜昌” ,” 巴中”, ” 池州”, ” 淮北”, ” 滁州”, ” 潮州” ,” 肇庆” ,” 芜湖”, ” 襄阳”, ” 赣州”, ” 鹤壁”, ” 龙岩”

- Cluster 4: ” 东莞” ,” 南昌”, ” 吉安”, ” 常州”, ” 广州” ,” 徐州” ,” 日照”, ” 枣庄”, ” 汉中”, ” 湖州”, ” 盐城”, ” 苏州”

- Cluster 5: ” 兰州” , ” 南通” , ” 台州” , ” 呼和浩特”, ” 昆明”, ” 济宁”, ” 淄博” , ” 深圳”

- Cluster 6: ” 成都” , ” 杭州”

- Cluster 7:” 大连”, ” 太原”, ” 武汉” ,” 郑州” ,” 青岛”

For cluster 6, these two cities both have many taxi companies and low entropy, what's more, there are many positive articles about taxies, which cause rating R is also small.

For cluster 7, these cities all have high entropy, more negtive events and more strict requirments than those of cities in cluster 6.

In cluster 2, cities have many news with topic 1, but in LDA-topic analysis we found that, topic 1 is hard to say positive or negtive. The ratings are small.

In cluster 1, many negtive news happened and ratings are larger. Cities in cluster 5 are normal city, because of 0 negtive articles, several positive articles, standard requirments and light monopolization.

5 Discussion

From the analysis in this report, I have some comments about this project, which may be useful for my further similar works.

- LDA-topic model can give out reasonable results especially when the articles are focusing on similar events or sharing same background. With detected topics and key words, Internet public sentiment monitoring system can be landed without too many other efforts.

- From the conclusion in clustering, we can find out that monopolization usually happened in middle cities, with some negtive accidents. This makes sense. Because of strength of monopolization power, government cannot control it efficiently resulting smaller ratings, which means relaxing the requirments to taxies. But many big cities have great competition in taxies market, which makes the quality of services better resulting in many positive events. In such cases, the government just needs to provide standard requirments, then the market itself will run in a good pattern.

- Clustering task for these cities can help people in relevant jobs save time. They do not need to analyze the monopolization and requirments in a city one by one. They just need to focus on several typical cities. And for government, this clustering result will benefit in discussing reasonability of relevant policies and making good issues or severe accidents under control.

- For further imporvement, larger data size will benefit a lot. Better word segmentation and more complicated method for calculating the sentiment score will also improve the result. Last but not least, clustering of bigger dataset may have more significant and credible patterns.