A Content-Based Recommendation Framework for Judicial Cases

Zichen Guo, Tieke He, Zemin Qin, Zicong Xie, and Jia Liu(✉)

State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China
liujia@nju.edu.cn

Abstract. Under the background of the Judicial Reform of China, big data of judicial cases are widely used to solve the problem of judicial research. Similarity analysis of judicial cases is the basis of wisdom judicature. In view of the necessity of getting rid of the ineffective information and extracting useful rules and conditions from the descriptive document, the analysis of Chinese judicial cases with a certain format is a big challenge. Hence, we propose a method that focuses on producing recommendations that are based on the content of judicial cases. Considering the particularity of Chinese language, we use "jieba" text segmentation to preprocess the cases. In view of the lack of labels of user interest and behavior, the proposed method considers the content information via adopting TF-IDF combined with LDA topic model, as opposed to the traditional methods such as CF (Collaborative Filtering Recommendations). Users are recommended to compute cosine similarity of cases in the same topic. In the experiments, we evaluate the performance of the proposed model on a given dataset of nearly 200,000 judicial cases. The experimental result reveals when the number of topics is around 80, the proposed method gets the best performance.

Keywords: Recommendation · Content-based · LDA · Cosine similarity

1 Introduction

With the development of computer science, it has been a very common ways to solute some difficult problems in reality by simulating with computer. Meanwhile, with the advancement of artificial intelligence, judicial judgement is getting closer to the justice of law with the aid of big data analysis. It is worth noting that the similarity analysis of judicial cases is the basis of wisdom judicature. A formative judicial case contains the court, the accuser and the accused, the fact, and the result of the case. In order to give credibility within a community, jury trials must take all these complicated factors into consideration with reference to similar cases. With the explosion in the number of judicial cases, it is difficult to consider similar cases without omission. Because of this, we seek to provide a novel recommendation method to assist judicial processing.

X. Cheng et al. (Eds.): ICPCSEE 2019, CCIS 1058, pp. 76–88, 2019.
https://doi.org/10.1007/978-981-15-0118-0_7

Starting with the study of Becker [1], researchers focus on what factors influence the optimal amount of enforcement, like the cost of catching criminals, the subjective decisions that affect the result. However, in practice, these factors are affected by political, moral and many other subjective constraints. Our main purpose is to make use of the objective factors among judicial cases.

Despite the fact that judicial study has gained some achievements in many aspects, such as legal word embeddings [2], inferring of the penalty [3] and judicial data standard [4], a recommender system is needed to deal with the large volume problem of judicial cases. In general, three filtering techniques such as content-based [5], collaborative [6,7] and hybrid filtering [8,9] are presented in the recommender system literature to filter records and identify the relevant information. Some of the progressive collaborative filtering algorithm [10,11] take cold start into consideration on the situation of lack of users or users' behaviours. In the meantime, it is challenging in judicial area because there exist many one-time users.

In view of the current situation, we propose an effective way to get recommendations, which is to collect the judicial cases a certain user put in. Our primary focus is to explore the judicial cases that are used to capture semantic similarities among text snippets. As mentioned above, given the cases that user input, the proposed model can return a recommended list of the relevant cases. We proposed our framework of content-based judicial case recommendation, as shown in a flow chart, Fig. 1.

In summary, we do the following work in this paper.

- We propose a content-based recommendation method for judicial cases.
- We develop a co-training process with TF-IDF and LDA to gain a plausible performance.
- We conduct an extensive experiments to test the performance of our proposed method, and the result reveals when the number of topic is around 80, our proposed method shows best performance.

The rest of this paper is organized as follows. Section 2 first describes relevant background of the models and algorithms, then sets out the proposed model and theoretical basis. Section 3 presents the experimental results and Sect. 4 summarizes this paper.

2 Methodology

2.1 Background

In this part, we provide detailed background of the models and algorithms used in this paper.

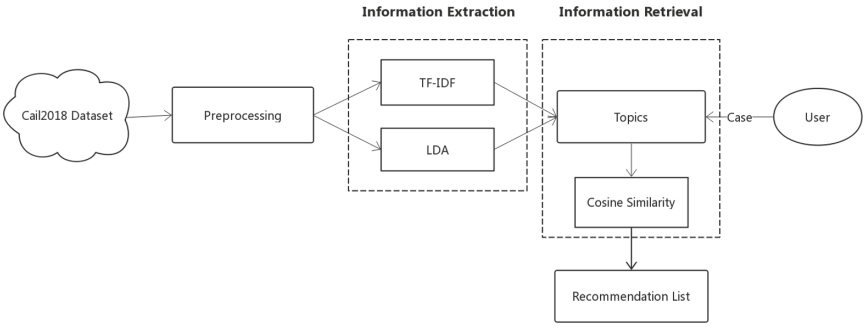

Fig. 1. Framework of content-based judicial case recommendation

Recommender Systems. Recommendation systems recommend items that specific users may be interested in books, news, movies, etc. At present, the methods of recommender systems are mainly based on collaborative filtering [12], association rules [13], content or hybrid algorithm [14]. LDA-based recommendation belongs to content-based recommendation.

Cold-starting is taken into consideration on the situation of lack of users or user behaviours and can be proved efficiently in many real projects. It also calls for attention in judicial field because existing a mass of one-time users.

Content-Based Recommendation with LDA. In natural language processing field, topic modeling is a kind of modeling for discovering the abstract "topics" that occur in a collection of documents. The LDA(Latent Dirichlet Allocation) model proposed by Blei in 2003 [19] has set the topic model on fire. The so-called generation model indicates that we think that every word in a document is achieved through the process of selecting a topic with a certain probability.

It's been a long time that LDA has been used to study user interests and build a system to recommend more friends with the same or similar user interests [17]. However, considering the lack of label of user interest and behavior among judicial cases, it is difficult to focus on user-generated content. We seek to turn to a new direction, which is to analyze and classify judicial cases input by users as content instead of user-generated content. In addition, TF-IDF is another reasonable algorithm in case recommendation.

TF-IDF Algorithm. TF-IDF is a commonly used weighting technology for information retrieval and data mining. TF means word frequency, IDF means inverse document frequency. TF-IDF proved useful and effective in stop-word filtering in various subject fields including text summarization and classification [18].

– TF Score (Term Frequency) considers documents as bag of words, agnostic to order of words. A document with 10 occurrences of the term is more relevant than a document with term frequency 1.
– We also want to use the frequency of the term in the collection for weighting and ranking. Rare terms are more informative than frequent terms. We want low positive weights for frequent terms and high weights for rare terms.

2.2 Preliminaries

For convenience, we define the custom data formats and definitions used in Table 1.

Table 1. Notations

Symbol	Description
M	Number of judicial cases
m	Index of a judicial case
N	Number of words in judicial case m
K	Number of topics
k	Index of a topic
R_m	Collection of words in judicial case m
c	Cause of action of a judicial case
q	Quantified data of a judicial case
l	Location of a judicial case
p	People involved of a judicial case
W_m	Collection of words in judicial case m
θ_m	Topic distribution of law case m
φ_k	Word distribution for topic k
$Z_{m,n}$	Topic assignment for $w_{m,n}$
$w_{m,n}$	The n-th word in case W_m

Definition 1 Judicial Case**.** A judicial case consists of a collection $R_m(c, q, l, p)$, which means that judicial case m is made up of the collections of words R_m with four elements c, q, l, p.

Definition 2 Topic**.** LDA defines each topic as a bag of words. Given a dataset of cases, topics maximize the posterior probability of the observed corpus.

2.3 Data Preprocessing

In light of the difference between Chinese and Romance languages, we use "jieba" text segmentation to get word sequences from dataset. For each judicial m in

the dataset, we get the collection $R_m(c, q, l, p)$. Also, a special filter is set up to filter out key data and sensitive vocabulary in the cases to remove interferences. We make a transformation $R_m(c, q, l, p) \rightarrow W_m(c)$ to get filtered collection of words in judicial case m.

2.4 Information Extraction

TF-IDF and LDA are trained to constitute the recommendation knowledge together in this part.

First, in order to smooth frequency of words in preprocessed data of M judicial cases, we use TF-IDF to obtain new corpus for the following training. TF-IDF assumes that if a word is important for a document, it would repeatedly appear in that document whereas it would be relatively rare in other documents. The TF is associated with the former assumption and the IDF is associated with the latter. TF-IDF is defined as

$$\text{tfidf}(t, d, D) = \text{tf}(t, d) \times \text{idf}(t, D)$$

where $f_{d(t)}$ is the normalized frequency of term $t \in w$ Therefore, it is defined as:

$$\text{tf}(t, d) = \frac{f_{d(t)}}{\max_{w \in d} f_{d(w)}}$$

In document d, $f_{d(t)}$ is the frequency of term t and w is an existing word. Also, $idf(t, D)$ shows the IDF t, which is defined as

$$\text{idf}(t, D) = \log_2\left(\frac{|p|}{|(d \in D, t \in d)|}\right)$$

where $|D|$ indicates the total number of documents in the corpus, and $|(d \in D, t \in d)|$ is the number of documents in which the term t appears.

The remaining words were filtered by frequency using the TF-IDF score. TF-IDF measures the importance of a word in a corpus as seen above. It increases with the number of occurrences in the document and decreases with the frequency in the corpus. We compute TF-IDF for each word of each document-plot in the corpus and keep a certain number of words with the highest score to optimize the corpus.

Although LDA assumes the documents to be in bag of words (bow) representation. We find success when using TF-IDF representation as it can be considered a weighted bag of words. It changes θ_m and φ_k in LDA model, as shown in Fig. 2.

We describe the LDA process of a judicial case data set in formal language, as shown below. Dirichlet() represents Dirichlet distribution and Multi() represents multinomial distribution.

1. For each topic $k \in 1, \ldots, K$, draw $\varphi_k \sim \text{Dirichlet}(\beta)$, denoting the specific word distribution for topic k.

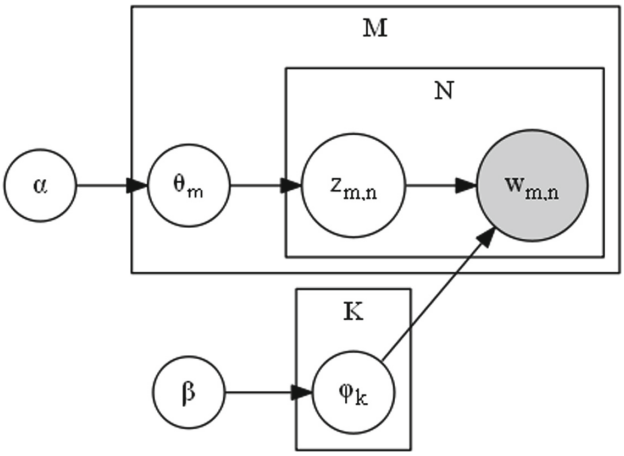

Fig. 2. Graphical representation of LDA model

2. For each judicial case $m \in 1, \ldots, M$:
   - Draw $\theta_m \sim$ Dirichlet(α), indicating the distribution of topics embedded in judicial case m;
   - For the n-th word in case m, $n \in 1, \ldots, N$, draw a W Multi(ϕ_z) for each word $w \in W_{m,n}(c)$.

The progress above can be used to gain knowledge among different kind of judicial cases. In order to generate recommendations for uses, we also need to do information retrieval from the topic distribution.

2.5 Information Retrieval

For each judicial case $m \in 1, \ldots, M$, we can get a vector of K topic distribution via information extraction, which is defined as

$$m = (s_1, \ldots, s_k)$$

where we seek s_i referring to the maximum among $s_1, \ldots, s_k$. On this occasion, i is the topic we regarded as the classification of case S. On account of two cases are similar if they contain similar topic contribution, similarity between cases is measured by cosine angle between vectors. Given a judicial case s input by user, which belongs to classification i, for each judicial case $\mathrm{t} \in 1, \ldots, M_i$, we get $\mathrm{Sim}(s, t)$, which is defined as:

$$\mathrm{Sim}(s,t) = \cos(s,t) = \frac{s \cdot t}{\|s\| \times \|t\|}$$

Recommendation list is composed of Top 5 cases of $\mathrm{Sim}(s, t)$.

3 Experiments

In this part, we give the whole realization of our framework.

3.1 Dataset

We perform experiments on the law case dataset CAIL2018_Small, which contains 204, 231 documents in total. After conducting TF-IDF, we retrieve a list of low value words (TF-IDF score under 0.025) and filter them out of the dictionary. In the end, we get a dictionary with 311, 024 words. Considering actual processing of judicial cases, we take a large number of judicial cases without manual labeling results into account. Therefore, we only consider using the fact description label in this dataset. In order to eliminate the interference items, we add the screening of time, place, person and number before data preprocessing, so as to get the final dataset. The specific methods for judicial cases are as follows:

- Regular expressions are used to match time keywords that appear in the cases.
- Regular expressions are used to match location keywords that appear in the cases, such as 'province', 'city', 'district'.
- Characters in the format of "XXX" are replaced by "PERSON" fields.
- For the regular matching of measurement units, the size of money is judged and divided into seven grades and marked as follows (Table 2):

Table 2. Measurement labels

Money range	Label
$[0, 10)$	$m1$
$[10, 100)$	$m2$
$[100, 1000)$	$m3$
$[1000, 10,000)$	$m4$
$[10,000, 100,000)$	$m5$
$[100,000, 1000,000)$	$m6$
$> 1000,000$	$m7$

To analyze the dataset as a whole, we give the statistics of money in the dataset, as shown in Fig. 3. Among the whole dataset, the proportion of Small-money criminal cases is very high, while the cases involving large amounts of money are very low. In all, the amount of m7-level criminal cases is 0. This figure reflects the case characteristics of CAIL2018_Small dataset from aspect of money. And the timeline of CAIL2018_Small dataset shows in Fig. 4.

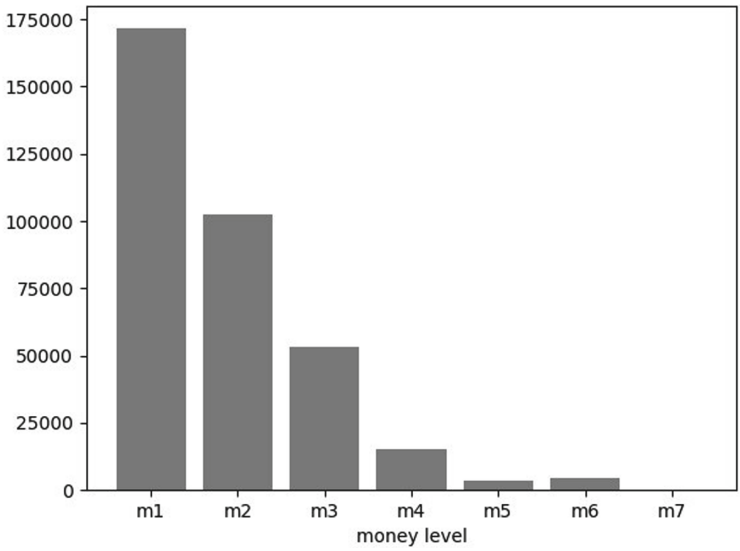

Fig. 3. Statistics of money

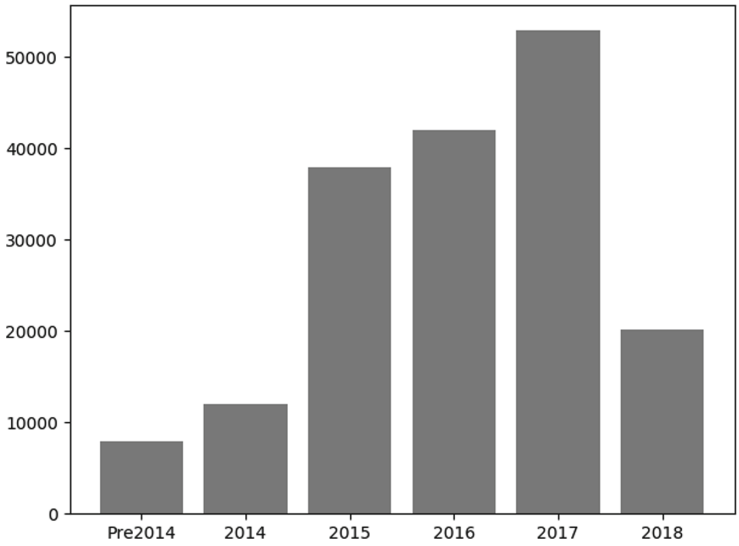

Fig. 4. Statistics of time

3.2 Experimental Results

We implement perplexity as the indicator [19]. Perplexity is a statistical measure of how well a probability model predicts a sample. In information theory, perplexity is the probability that the test data is monotonically decreasing, which

is the algebraic equivalent of the inverse of the probability geometric mean of each word. The lower the complexity score, the better the generalization performance [20]. Perplexity of the untrained dataset (D_{test}) is defined as follows:

$$\text{perplexity}(D_{\text{test}}) = \exp\left(\frac{-\sum_{d=1}^{M} \log(p(w_d))}{\sum_{d=1}^{M} N_d}\right)$$

where M is the total number of documents in judicial dataset. In document d, W_d represents words and N_d is the number of words.

Among the primary setting, for each num of topic $k \in [10, 150]$, we set hyperparameters $\alpha = \frac{50}{k}$, $\beta = 0.01$, following the studies of [21]. Figure 5 illustrates the perplexity figures with different numbers of topic k.

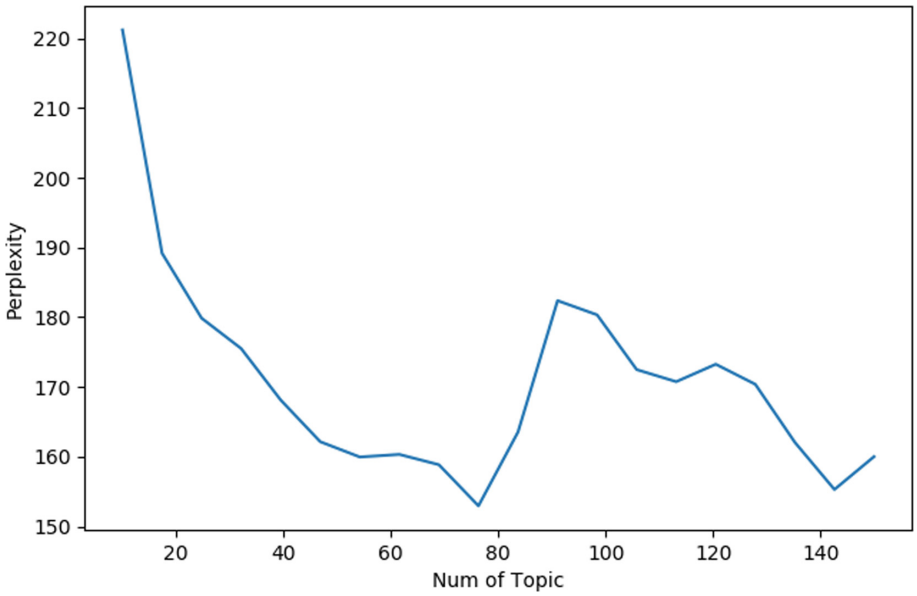

Fig. 5. Results of k-topic LDA model with TF-IDF in perplexity

As can be seen in Fig. 5, when num of topic $k \simeq 80$, perplexity requires the minimum value about 155, which is acceptable. The perplexity declines significantly when $k \in [10, 50]$, and are in an upward trend when $k \in [80, 95]$, but also generally falls for $k > 95$ in the process.

Next we figure out exactly the value of k, we reduce the scope and choose $k = 75$, 76, 77, 78, 79, 80, then calculate the perplexity as showing in Fig. 6.

As shown in Fig. 6, when $k = 78$, perplexity achieves the minimum value nearly 154. In all, we choose $k = 78$ as ideal topic number. We display the top 30 words with TF-IDF value in the model with $k = 78$, as shown in Fig. 7.

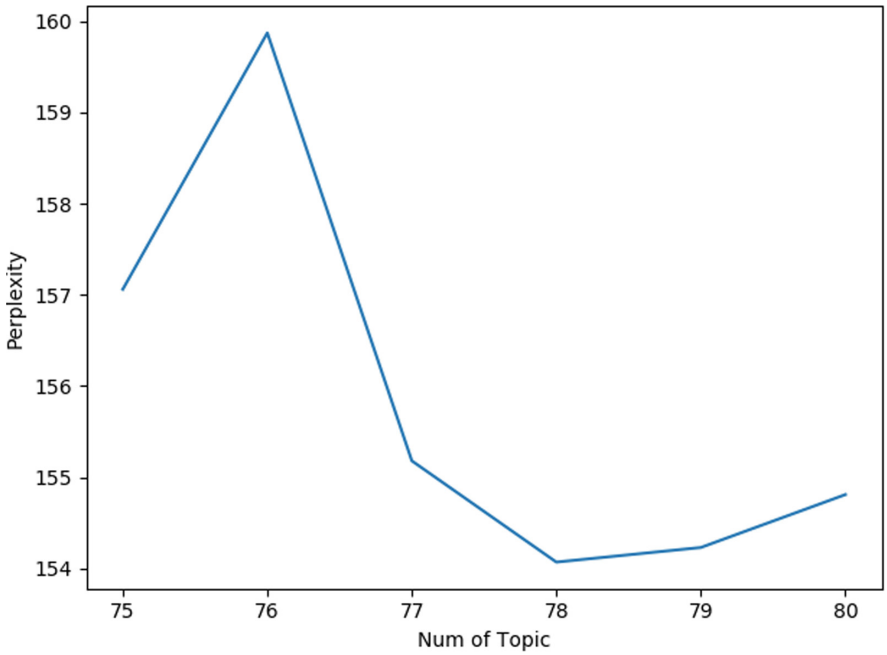

Fig. 6. Results of perplexity $k \in [75, 80]$

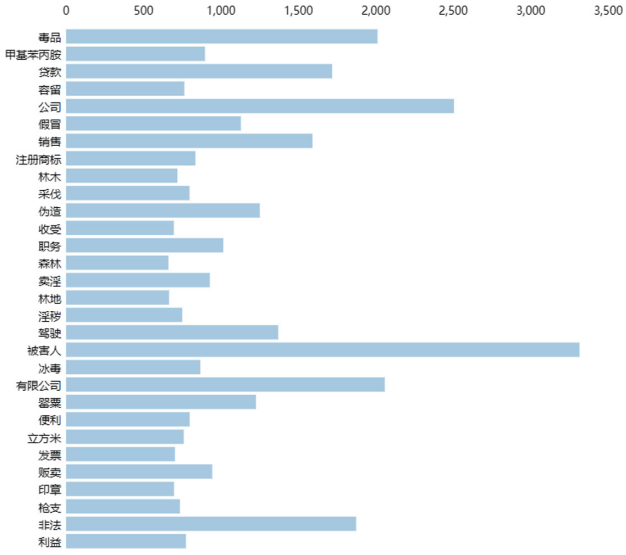

Fig. 7. Top 30 words

In order to test the actual result of our model, we simulate a series of tests to show model's performance. Firstly, we build a classified corpus according to the topic distribution of each document in CAIL2018_Small dataset. More specifically, for each document, we choose most probable topic as its subject catalog. After this, we build matrix similarity indexes for each topic catalog. After classifying corpus, we can recommend cases to users. Here, the experiment simulates judicial cases input by user. For example, a user enters judicial case as follow (Fig. 8):

2017年10月中旬，被告人于伟仁联系刘海明欲购买甲基苯丙胺（冰毒）2000克，刘海明遂联系李伟寻找毒品卖家。10月20日左右，李伟、王少成（刑拘在逃）分别联系王少廷购买甲基苯丙胺，其中王少成联系购买1000克，李伟联系购买2000克。王少廷遂联系范敬仰购买甲基苯丙胺3000克，范敬仰又联系张高阳购买3000克甲基苯丙胺。后经联系约定，范敬仰出售给王少廷甲基苯丙胺每克100元，王少廷出售甲基苯丙胺每克110元，刘海明告知于伟仁甲基苯丙胺每克125元。10月22日早上6时许，张高阳、范敬仰、王少廷、李伟、刘海明、于伟仁、王少成等人相互电话联系询问毒资情况并约定进行毒品交易。李伟电话告知王少廷让刘海明和于伟仁带着毒资去姜寨镇，上午10点左右到姜寨镇见面交易。王少成联系王少廷于10点左右到庞营韩老家集上取毒资，后王少廷找到王少成拿到毒资10万元，两人约定剩余1万元毒资等毒品出售后再补足，王少廷携带该10万元毒资回到姜寨镇玉楼村与前来购买毒品的刘海明、于伟仁见面，于伟仁称毒资在随身携带的银行卡上，问王少廷是否能转账付钱，王少廷电话联系范敬仰后不同意转账交易，刘海明、于伟仁便去银行取款，于伟仁到银行取出59000元现金。王少廷找到范敬仰交10万元毒资，后范敬仰携带该毒资到张高阳家交易毒品，张高阳将一包毒品交给范敬仰。范敬仰携带该毒品回到家中交给王少廷。王少廷将毒品放在出租车内，开车带着范敬仰返回姜寨准备将毒品交给王少成。在姜寨镇后王楼村，民警将王少廷、范敬仰抓获，当场从其驾驶的出租车后排座上查获一包嫌疑毒品。刘海明、于伟仁计议前往临泉县城继续用银行卡取款，在前往临泉县城的路上被公安机关查获，民警从后排座上查获现金59000元。经称量，嫌疑毒品净重994.61克；经鉴定检出甲基苯丙胺成分，含量为79.6%。侦查机关当日将张高阳抓获，12月13日将李伟抓获。

Fig. 8. Case input by user

推荐案例170536 相似度：0.961252
公诉机关指控：被告人范某某与张某（另案处理）系同居男女朋友关系，被告人范某某明知张某实施贩卖毒品行为，仍用自己的银行账户，帮助张某保管毒赃、提取毒资。具体事实如下：1、2015年7月底至11月期间，张某雇佣余某（另案处理）去广东购买1000克甲基苯丙胺毒品用于贩卖，后在温岭市、台州市椒江区等地将该1000克甲基苯丙胺出卖给曹某、王某1等人，其中王某1于2016年9月初至10月期间，先后汇到被告人范某某持有的账户尾号为4925的农村信用社储蓄卡毒赃人民币共计47500元。2、2015年10月4日，被告人范某某受张某指使，至中国农业银行玉环县清港支行给帮助张某购买毒品的余某控制的银行账户汇去毒资人民币35000元。2015年11月28日上午，温岭市公安局禁毒大队民警在玉环县清港镇下湫村八片区0066号出租房抓获被告人范某某。被告人范某某归案后，如实供述主要涉案事实。

推荐案例137726 相似度：0.949229
上海市宝山区人民检察院指控：2015年8月13日19时许，被告人牟某某携带18张他人银行卡至上海市闸北区秣陵路XXX号火车站邮政支局门口，欲出售牟利时被民警抓获，同时民警缴获其随身携带的上述银行卡。被告人牟某某到案后如实供述上述犯罪事实。

推荐案例160012 相似度：0.946199
湖北省武汉市武昌区人民检察院指控：被告人张1某于2016年10月21日16时许，应周某购买毒品的要求，在武汉市武昌区武泰闸栅栏口与周某相约见面，收取周某的购毒款人民币1100元，并与周某约定稍后在武汉市武昌区南湖社区雅安街南嘉宾馆向周某交付毒品。随后，被告人张1某以人民币1000元的价格向他人购得塑料袋装白色晶体颗粒毒品1包及红色片剂毒品3颗，并从白色晶体颗粒毒品中分装少许留作其个人吸食。当日17时许，被告人张1某在武汉市武昌区南湖社区雅安街南嘉宾馆欲向周某交付毒品时，被公安民警当场抓获。公安民警当场从其身上查获上述毒品。经称量、取样及鉴定，所查获的毒品均为甲基苯丙胺，共计净重10.37克。指控上述事实，公诉机关提供有下列证据予以证实：1、公安机关出具的抓获经过、破案经过；2、物证（扣押清单及照片）；3、证人周某的证言；4、被告人张1某的供述与辩解；5、称量、取样笔录及鉴定材料。公诉机关认为，被告人张1某违反国家毒品××法规，以牟利为目的为他人代购毒品，其行为触犯了《中华人民共和国刑法》×××、××的规定，已构成贩卖毒品罪。提请本院依法追究其刑事责任。

Fig. 9. Recommended case

Then we load the topic index, calculate the similarity between the input case and each cases in the indexcatalog by cosine similarity. We select the top 5 cases of similarity as the recommendation judicial cases to present to the user. Top three judicial cases is shown in Fig. 9 and the cosine similarities are 0.9613, 0.9492, 0.9462.

4 Conclusion

In this paper, we present a content-based method of judicial case recommendation to address the problem of how to help user better understand judicial cases in depth. Specifically, we develop a co-training process with TF-IDF and LDA to gain a plausible model performance. Given LDA is an unsupervised learning algorithm, we conduct experiments to evaluate the performance of the proposed recommender system. The results show the optimal number of topic. Our recommendation method still has some room for improvement. Putting state-of-the-art algorithms into practice with good performance is always a critical problem, which we will focus on in the future.

Acknowledgment. The work is supported in part by the National Key Research and Development Program of China (2016YFC0800805) and the National Natural Science Foundation of China (61772014).

References

1. Becker, G.S., Landes, W.M.: Essays in the Economics of Crime and Punishment. Number 3 in Human Behavior and Social Institutions. National Bureau of Economic Research: Distributed by Columbia University Press
2. He, T., Lian, H., Qin, Z., Zou, Z., Luo, B.: Word embedding based document similarity for the inferring of penalty. In: Meng, X., Li, R., Wang, K., Niu, B., Wang, X., Zhao, G. (eds.) WISA 2018. LNCS, vol. 11242, pp. 240–251. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-02934-0_22
3. He, T.-K., Lian, H., Qin, Z.-M., Chen, Z.-Y., Luo, B.: PTM: a topic model for the inferring of the penalty. J. Comput. Sci. Technol. **33**(4), 756–767 (2018)
4. Qin, Z., He, T., Lian, H., Tian, Y., Liu, J.: Research on judicial data standard. In: 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), pp. 175–177. IEEE (2018)
5. Balabanovic, M., Shoham, Y.: Fab: content-based, collaborative recommendation. Commun. ACM **40**, 66–72 (1997)
6. Linden, G., Smith, B., York, J.: Amazon.com recommendations: item-to-item collaborative filtering
7. Das, A.S., Datar, M., Garg, A., Rajaram, S.: Google news personalization: scalable online collaborative filtering. In: Proceedings of the 16th International Conference on World Wide Web - WWW 2007, p. 271. ACM Press (2007)
8. Badaro, G., Hajj, H., El-Hajj, W., Nachman, L.: A hybrid approach with collaborative filtering for recommender systems. In: 2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC), pp. 349–354, July 2013

9. Strub, F., Mary, J., Gaudel, R.: Hybrid collaborative filtering with autoencoders (2016)
10. Ahn, H.J.: A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. Inf. Sci. **178**(1), 37–51 (2008)
11. Patra, B.Kr., Launonen, R., Ollikainen, V., Nandi, S.: A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data. Knowl.-Based Syst. **82**(C), 163–177 (2015)
12. Ekstrand, M.D.: Collaborative filtering recommender systems **4**(2), 81–173
13. Lin, W., Alvarez, S.A., Ruiz, C.: Efficient adaptive-support association rule mining for recommender systems. Data Min. Knowl. Disc. **6**(1), 83–105 (2002)
14. Kardan, A.A., Ebrahimi, M.: A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups. Inf. Sci. **219**, 93–110 (2013)
15. Nagori, R., Aghila, G.: LDA based integrated document recommendation model for e-learning systems, pp. 230–233, April 2011
16. Luostarinen, T., Kohonen, O.: Using topic models in content-based news recommender systems. In: Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013), pp. 239–251. Linköping University Electronic Press, Sweden (2013)
17. Pennacchiotti, M., Gurumurthy, S.: Investigating topic models for social media user recommendation. In: Proceedings of the 20th International Conference Companion on World Wide Web, WWW 2011, pp. 101–102. ACM, New York (2011)
18. Ramos, J.: Using TF-IDF to determine word relevance in document queries
19. Blei, D.M.: Latent Dirichlet allocation, p. 30
20. Arora, K.: Contrastive perplexity: a new evaluation metric for sentence level language models. CoRR, abs/1601.00248 (2016)
21. Yin, H., Sun, Y., Cui, B., Hu, Z., Chen, L.: LCARS: a location-content-aware recommender system, pp. 221–229, August 2013