Intention Classification Of Customer Service Dialog

GUANNAN LU,20454477, Hong Kong University of Science and Technology

KEYWORDS
Intention, Response Cluster, RNN

Abstract

Intention understanding of customers' request is an essential part for an artificial intelligence customer service dialogue system. The footstone of this system is the understanding of request in one single turn dialog. In the traditional way, the service' response is determined by key words matching and is useless when the key words of questions have never been included, even if the meaning is the same to one that already exists in database. Now, by mining customer service records and training a RNN network, we can easily classify the request into one particular intention. However, there two main challenges: 1. time-consuming work when tagging thousands of requests; 2. hard to improve the performance of prediction with high-dimension data and long sequence. This paper addresses the above issues by clustering the response as the tag of request firstly and proposing an architecture using bidirectional LSTM network with timedistributed layer. In our customer service dialog of China Mobile, the same intention of different request does have a similar set of words of the response from services. The experiments on China Mobile customer service dialog data show that cluster similar response will greatly speed up the process of tagging and get a set of similar requests with different words and format. And, the results of predicting customers' intention show this network architecture can effectively extract semantics.

1. Introduction

Nowadays, natural language processing has become a hot topic in artificial intelligence and a potential application domain. The intelligent customer service dialogue system can quickly reply questions, accurately locate users' issues, save customer service seats, and reduce training costs. Smart customer service can provide quick and unified answers to key and hot issues to ensure service standardization. In this scene, intention classification is an groundwork to multi−turn dialog system. By analysis the intention exists in users' question, the service could execute corresponding action and return proper answers.

2. Data Preparing

2.1 Data Description

Data comes from the customer service dialog of china mobile, which has the following columns: 'sessionid','id','acceptnumber','requesttime','request','serviceid','responsetime','response'

Each sample looks like:

	sessionid	id	acceptnumber	requesttime	request	serviceid	responsetime	response
10	13410000258T16081911402229APP	73	10000000258	31DEC99:00:00:00	-1	SZ31537	19AUG16:11:46:07	好的
11	13410000258T16081911402229APP	74	10000000258	31DEC99:00:00:00	-1	SZ31537	19AUG16:11:46:17	如果没有其他业务，小和先退下了，收到10086046短信后请回复数字1，赏赐小和本人1008...
12	13410000258T16081911402229APP	75	10000000258	19AUG16:11:46:43	可以了谢谢	SZ31537	31DEC99:00:00:00	-1
13	13410000258T16081911402229APP	76	10000000258	31DEC99:00:00:00	-1	SZ31537	19AUG16:11:46:51	不客气
14	13410000422T16082318212113APP	118	10000000422	23AUG16:18:21:25	人工	SZ31908	31DEC99:00:00:00	-1
15	13410000422T16082318212113APP	119	10000000422	23AUG16:18:21:26	Hello	SZ31908	31DEC99:00:00:00	-1

graph 1 Data sample

2.2 Data Clean

In order to feed the classification model, we need to separate the dialog into single turn pair and gather request and response in one row. More importantly, trash message and no semantic sentence should be removed.

2.3 Segmentation And Word Embedding

There are two methods of word embedding we did. The first one is using a pre-trained model for Chinese word embedding which we can get from the Internet.The second one is using our own corpus to train a new word embedding model.

Compared with the first method, the second one shows a more accurate embedding as the corpus used for training is more related to the context of this job.

2.4 Sentence Embedding

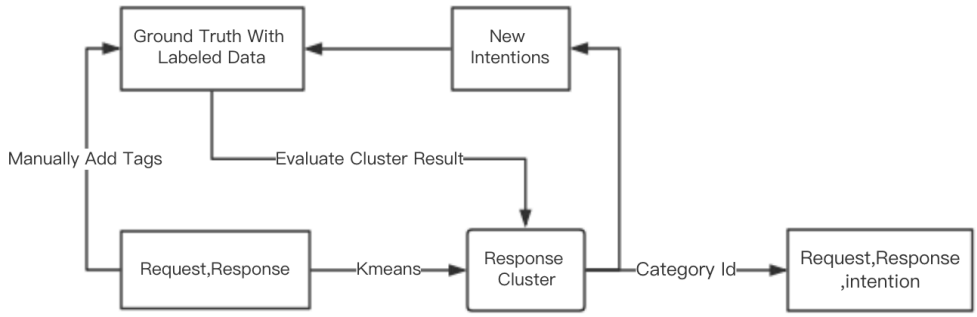
After word embedding, we need to embed a sentence since both requests and responses are sentences.

There also exists several methods for sentence embedding. The first and most traditional way is using the average value of all the word vectors in the sentence. It is computationally cheap and easy to implement

We also try some other ways. One is using a recurrent neural network (LSTM) to train a model whose input is the word vectors by word2vec and output is the word vectors by one-hot encoding. This model try to discover the relation between words and sentences. After training, we can use this RNN to implement sentence embedding. We called it encoder-decoder model.

The other one is using weighted average rather than pure average. As we have built a key word list which contains some important word of the business. We can put more weight on these words.
The last one is using IDF for sentence embedding. In this way, we give more weights on the words that have larger IDF.

2.5 Clustering Of Response And Return Tag



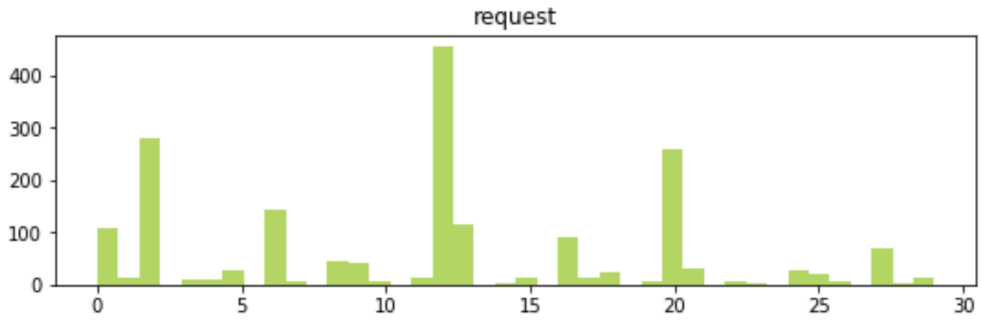
graph 2 Labeled Data Generator

This structure will generate more and more labeled data iteratively. Unlabeled data will be clustered to get several categories and we could regard the category id as a kind of intention. If a category has some common meaning, a new intention will be found and added into the intentions set. As a result, these data with new intention will rich the ground truth data set created manually, which will evaluate the result of cluster by adjusted Rand index.

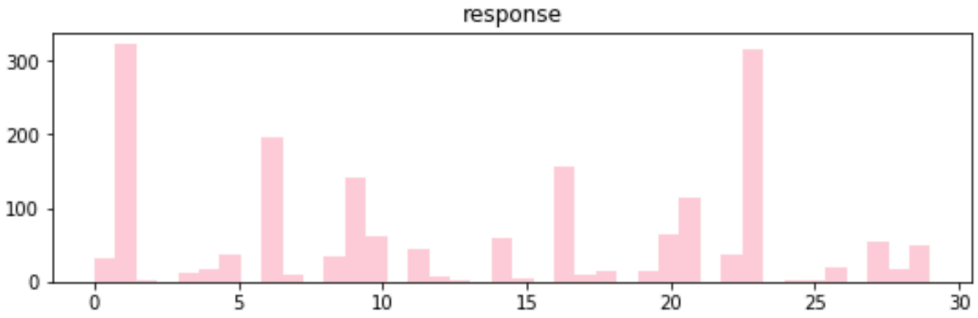
The Rand index or Rand measure (named after William M. Rand) in statistics, and in particular in data clustering, is a measure of the similarity between two data clusterings. A form of the Rand index may be defined that is adjusted for the chance grouping of elements, this is the adjusted Rand index.

2.5.1 Cluster Method Comparison

In order to get more labeled data quickly, it is an essential step to cluster response rather than request. Below are the comparison of cluster result by using different clustering sources.



graph 3 Cluster on request



graph 4 Cluster on response

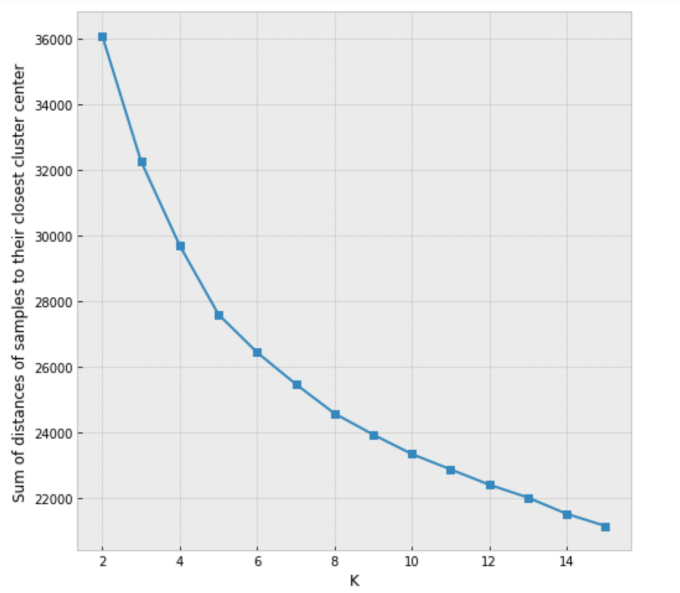
From the graphs above, it is obviously that clustering based on response will get a more balanced result than the other, which means that more categories can be easily separated and prepare a good footstone to intention classification.

We tried several methods, include agglomerativeClustering, kmeans and dbscan, to cluster the response. In semantic understanding by our review, Kmeans get the best classification result.

```
from sklearn.cluster import KMeans,DBSCAN,AgglomerativeClustering
# kmeans
k = 7
clf = KMeans(n_clusters=k) #设定k, 这里就是调用KMeans算法
s = clf.fit(sens_vec) #加载数据集合

# dbscan
# clf = DBSCAN(eps=1.27, min_samples=20)
# clf.fit(sens_vec)

# AgglomerativeClustering
# clf = AgglomerativeClustering(n_clusters=7)
# clf.fit(sens_vec)

numSamples = len(sens_vec)
centroids = clf.labels_
# print(centroids) #显示中心点
# print(clf.inertia_)  #显示聚类效果
# print(clf.cluster_centers_) #簇的中心向量
pd.DataFrame(list(clf.labels_))[0].value_counts()
```

graph 5 Cluster methods

For Kmeans, we need to define the proper cluster numbers. From the graph below, where the vertical coordinates means sum of distances of samples to their closest cluster center, we know that number around 7 is proper.



graph 6 sum of distances of samples to their closest cluster center

2.5.2 Cluster Result

Let's take a look at several cluster result:

Cluster 0: mainly about take some telecoms package

72	13410015799T16080109540760APP	办理流量叠加包	办理多少钱的流量叠加包	0
89	13410020336T16081511415680APP	那要怎么办理	回复WHXD2申请15G流量年包优惠本机需要改为58元套餐	0
90	13410020336T16081516445077APP	办理流量年包办理流量年包	本机可以参加，需要承诺连续12个月使用58或以上4G套餐获得年包优惠， 意思就是自己使用...	0

Cluster 2: mainly about canceling the telecoms package

	sessionid	request	response	cluster
4	13410001228T160803125048	是的，我要求取消的昨天你们客服打电话来，我没接到	显示订单回退了，就是承诺使用38元套餐2个月的那个订单	2
13	13410001228T160807211011	能否帮我查下现在用的是不是六块月租的	，您现在是6元的月租的轻松卡的套餐的	2
41	13410008121T160807120304	那可以升级19元吗	指的是套餐要改为19元流量王套餐吗?	2
63	13410013735T160818151135	更改到哪个取消50元流量套餐	最低10元畅听卡哦	2
77	13410017283T160817092455	没有呢套餐怎么取消	4G套餐有4G主体/非4G主体套餐。其中4G主体套餐又分语音类（适用手机）和数据终端类（适用...	2
78	13410017283T160817092455	4G上网套餐的取消方法	4G上网套餐已停止办理，取消后不能重新开通，如需确实需取消，可通过套餐转换形式转为办理4G飞...	2

Cluster 3:mainly about asking more information about the telecoms package

161	13410035760T16080509403099APP	套餐内容你发过来我看看	好的，请稍等【套餐】38元4G飞享套餐【月费】38元/月，送来电显示，国内接听免费【流量】含...	3
162	13410035760T16080509403099APP	还有其他套餐吗便宜点的	【套餐】18元4G飞享套餐【月费】18元/月，送来电显示，国内接听免费【流量】100M国内手...	3
174	13410039261T16082112220330APP	全球通有些套餐	，全球通主要是4G飞享套餐家族共有14个成员：18元、28元、38元、48元、58元、 88...	3
175	13410039261T16082112220330APP	套餐详情	要全部的吗【套餐】18元4G飞享套餐【月费】18元/月，送来电显示，国内接听免费【流量】10...	3

Cluster 6: mainly about the Mobile traffic

11	13410001228T160807211011	流量不清零的介绍	餐（不含4G随心王）、手机流量套餐、闲时套餐，当...	6
19	13410002616T16082008352093APP	WLAN流量是什么	就是移动公众场所的WIFI	6
44	13410008121T160807120304	我说的是奥运流量包19元hello	只能尝试发送19元 发送KTAYLLB19至10086。办理	6
53	13410008884T16081619464102APP	通用流量还是定向流量呢是不是什么方面都可以用的呀	，不是的哦会员权益流量包的使用方法：央视咪咕视讯会员、新媒体会员客户可通过线上办理页面、短信...	6

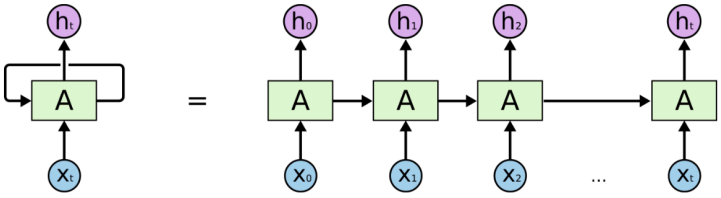
3. Intention Classification with Bidirectional LSTM Network of TimeDistributed Layer

LSTM is an excellent variant model of RNN. It inherits most of the characteristics of RNN model, and solves the problem of Vanishing Gradient caused by the gradual reduction of gradient in back-propagation process.

The bidirectional LSTM is an extension of the traditional LSTM and can improve the model performance of the sequence classification problem. In some problems where all the input of sequence are available, the bidirectional LSTM trains two instead of one LSTM on the input sequence. The first of the input sequences is the original in forward direction, and the second is the inverted copy of the input sequence which is in backward direction. This can provide the network with additional context and can understand the problem faster and more comprehensively.

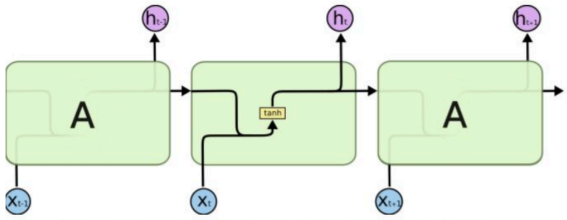
3.1 LSTM For Long Sequence

A LSTM network is a kind of recurrent neural network. A recurrent neural network is a neural network that attempts to model time or sequence dependent behaviour – such as language, stock prices, electricity demand and so on. This is performed by feeding back the output of a neural network layer at time t to the input of the same network layer at time t + 1. It looks like this:

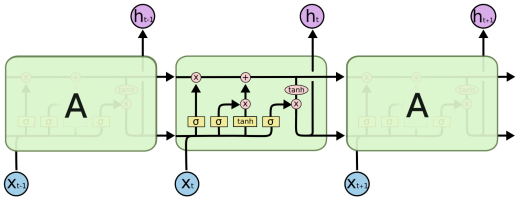


An unrolled recurrent neural network

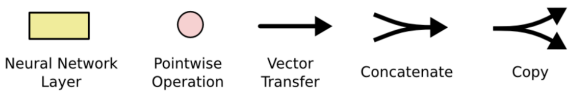
LSTM (Long Short Term Memory Networks) is a special RNN that can learn and store long-term dependencies. It has the characteristics of a recurrent neural network: repeated chained network storage structures. The graphs below are the different between RNN and LSTM network.



RNN Network



LSTM Network



Descriptions of LSTM Components

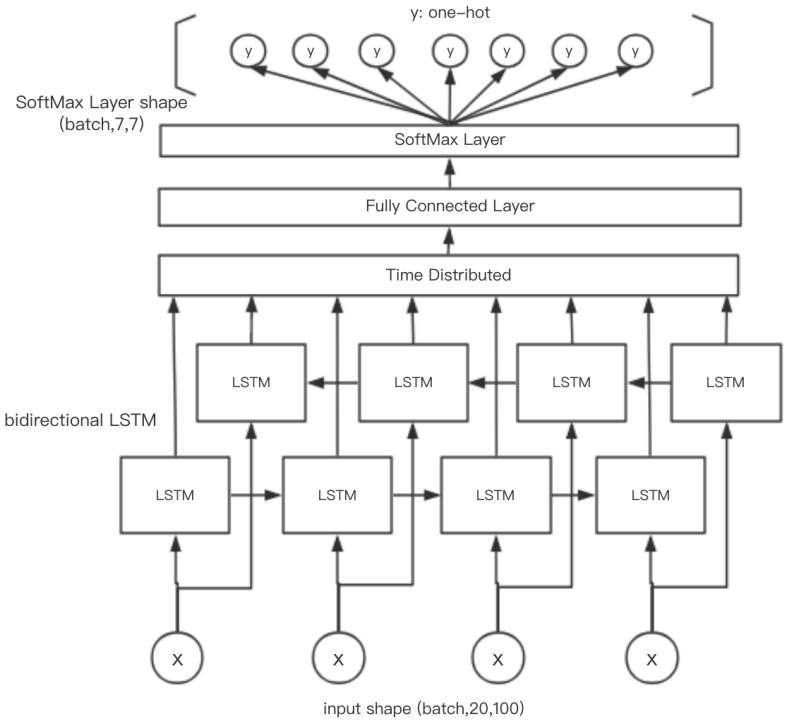
In the graph above, each row contains a complete vector, from the output of a node to the input of other nodes. Pink circles denote punctuate operations, such as vector addition, while yellow boxes represent learning neural network layers. Line merging means concatenation, while a branch indicates that its contents are being copied and copies will be transferred to different locations.

The key to LSTMs is the cell state, which is the horizontal line running through the top of the chart. It runs through the whole chain, with only some minor linear interactions. Information is easy to

flow in a constant way. LSTM has the ability to remove or increase information to cell state by elaborately designed structure called "gate". The door is a way to allow information to be passed. They contain a sigmoid neural network layer and a pointwise multiplication operation.

3.2 Bidirectional LSTM Network of TimeDistributed Layer

In our project, we design a network structure with a bidirectional LSTM layers and one time distributed layer created by using Keras, which is a deep learning framework based on Theano. The graph below is the structure of our model from word embedding input to prediction vector output. The length of X is 20, which is the maximum number contains in one sentence. And every X is a word2vec embedding vector of 100 dimensions. After calculated by both forward propagated and backward propagated lstm layers, fully connected and softmax layers will choose one most possible intention.



graph 7 Model structure

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_7 (Embedding)      (None, 20, 100)           252400
_________________________________________________________________
bidirectional_7 (Bidirection (None, 20, 50)            25200
_________________________________________________________________
time_distributed_7 (TimeDist (None, 20, 24)            1224
_________________________________________________________________
flatten_6 (Flatten)          (None, 480)               0
_________________________________________________________________
dense_20 (Dense)             (None, 14)                6734
_________________________________________________________________
dense_21 (Dense)             (None, 7)                 105
=================================================================
Total params: 285,663
Trainable params: 33,263
Non-trainable params: 252,400
```

graph 8 Parameters in model

The corresponding codes of this structure are as the graph below.

```
model = Sequential()
vocab_size = len(vec)
model.add(Embedding(vocab_size, 100, weights=[vec], input_length=20, trainable=False))
model.add(Bidirectional(LSTM(25,dropout=0.4,return_sequences=True)))
model.add(TimeDistributed(Dense(24, activation='relu')))
model.add(Flatten())
model.add(Dense(14, activation='sigmoid'))
model.add(Dense(7, activation='softmax'))
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['categorical_a
ccuracy'])
```

graph 9 Codes to building network structure o

Embedding layer use a pre-trained word2vec to represent the words in a space with a high dimension. “Input_length” is the fixed length of words in one sentence and variable “vec” is the embedding mapping matrix, each vector in matrix has 100 dimensions. This means that the output of the embedding layer will be a 3D tensor of shape (batch, sequence_length, embedding_dim).

Bidirectional LSTMs are supported in Keras via the Bidirectional layer wrapper. Propagating in both direction will catch more message in sentence.

The TimeDistributedDense layers allows you to apply that Dense function across every output over time. When you apply a TimeDistributedDense, you're applying a Dense layer on each time step, which means you're applying a Dense layer on each h1, h2,...,ht respectively. This is important because it needs to be the same dense function applied at every time step. If you didn't not use this, you would only have one final output - and so you use a normal dense layer. This means you are doing either a one-to-one or a many-to-one network, since there will only be one dense layer for the output. And another reason to do is that each output of one RNN cell is a 3D vector which is different from the 2D vector the dense layer needs, in case we have to flatten them together.

The next layer is called flatten layer which gather the output together. These vector will be put into a fully connected layer with 14 cells. Lastly, softmax layer will pick up one intention index in a one-hot vector with the highest possibility.

4. Experiment And Result

Sentence Embedding Method	Training Precision	Testing Precision
Baseline model (Average)	0.441	0.372
Weighted Average based on IDF	0.625	0.586
Weighted Average based on Keywords	0.674	0.647
Encoder-decoder model	0.705	0.709

The following table shows the result of intention prediction by LSTM model. By using different sentence embedding methods, the accuracy rise up slowly. So, our model can handle this task.

5. Conclusion

This article proposes a response clustering method and bidirectional lstm model to speed up and well predict users' intention, which can be quickly transfer to other domain using different dialog records. It is a potential way to get thousands of labeled data in a very short time by regards the response clustering result as an category of intention. Weighted average based on keywords works well to map the sentence into matrix. The lstm model also get a good result on accuracy.

6. Future Work

In the future, we may work more on intention detection in multi-turn dialog. There are many useful message in context, however, we just divide them into separate single turn dialog and lost some information. It will be greatly improve the accuracy of intention detection if we use the context of dialog.

Attachment: Meeting minutes

Date: Feb 7
Participants: Prof. Fangzhen Lin, all registers of the project. Content:

- Introduction and intuition of the project
- Data source introduction and preview
- Q&A session.

Date: Feb 28
Participants: Prof. Fangzhen Lin, Huang Yilun, Lu Guannan, Mi Lan
Content:

- Project details discussions, including each steps of user intention classification, model candidates, automatic tagging using clustering
- Feedback from Prof.Lin: take rules into consideration, learn from IBM knowledge grapf building process, ensemble learning.

Date: Apr 9
Participants: Prof. Fangzhen Lin, Xiao I Robot, Lu Guannan, Mi Lan
Content:

- Middle report of our project.
- Prof.Lin and Xiao I Robot raised question about data cleaning and automatic tagging part of our project.
- Xiao I Robot introduced their methods to solve this problem in current stage

Date: May 11
Participants: Prof. Fangzhen Lin, Huang Yilun, Lu Guannan, Mi Lan Content:

- Report the progress of the project. It mainly includes four parts: data preprocessing, word/sentence embedding, clustering, classification. We show our results of part 1-3. Prof. Lin carry out some ideas and questions of our solution.
- Discuss about how to enhance the performance of clustering (labeling).
- Prof. Lin emphasize the classification is the main part of the project. We should pay more attention on it.