A Memory-based Hybrid Architecture for Online Customer Services

Liang Xu, Xiaosheng Dai
Fitme.ai
{xul,dxs}@fitme.ai

Abstract

A good agent of online customer services should have the ability to interact with users in a successive way and be able to answer questions of users with high accuracy. In this work, we explore this direction by designing an architecture that fuse a memory-based multiple model and a single turn model. Furthermore we used unsupervised learning method to reduce search space in several order of magnitudes, and used large external corpus to generate word vector to capture rich linguistic information to boost performance. As a result, our approach yields a very good result. It clearly demonstrate the ability to conduct successive interaction, with 83.3% accuracy in single-turn in our test. To our knowledge, this is the first time a memory based hybrid approach used in online customer services.

1. Introduction

Bot Customer Services is very common application in many industries nowadays. By utilizing bot to do online customer services, labor cost can be reduced and workload of customer service staffs can be lightened. Bot Customer Services is a also typical application of NLP. Online customer raises a question related to certain business, an agent will try to answer the question that aim to help the customer. Although this kind of scenario is quite simple, to design a practical and efficient agent is still difficult. One reason of this is that human language is ambiguous in many cases, as same sentences can represent different meanings. Another reason is that usually the best response should take context into consideration. To model the context of multiple turns of dialogue is still a challenge task currently. It needs tremendous dialogue corpus of training data as problem become more complicated when involve context. But multiple turns dialogues corpus in close domain is always scarce and limited. Additionally, there is one practical issue, even some sophisticated models can be applied to model context, but the ability of retrieving the most relevant answers in single turn for these kinds of models is weakened.

To tackle above problems, we utilize an architecture that combine a multiple turns model and a single-turn model. Our proposal mix model has advantage that able to model context in question answering, while efficiently retrieve the most relevant answer for query. Except new architecture we've designed, we also made three major modifications aiming to improve the quality of our system:

Firstly, deep learning methods typically require huge amount of data. As we discussed above, dialogue corpus in specific domain is limited even in big industrial companies. However, raw text corpus like news and knowledge question answering is abundant. Thus we use pre-trained language model in huge raw text corpus with word2vec to learn semantic representation for each

word. And then it can be used to boost performance.
Secondly, unsupervised learning like clustering is useful for some supervised learning tasks. By doing cluster, we can learn some internal structure from the data. Possible answers from dialogue history collected from staff services are huge. Even for some questions, different staff services may have many different answers with the same contexts. Therefore, we utilize clustering for answers to reduce possible space. Similar answers are clustered to same group, and it will be represented by one answer that picked by staff services.
Thirdly, value network is used to rank among answer candidates. It tries to distinguish between more related answer from less related answer for a question. we use value network to choose the best answer from top-k possible answers.

2. Related Work

Memory based network, where our model based from, is a kind of models that has an trainable memory to store information to and retrieve from. It can be used to do inference and reasoning, question answering, sentiment analysis and so on. We will brief introduce some of development of memory based models and related data set.
To our knowledge, memory network(Jason et al., 2014) is the first model that introduce memory based network. It consists of a memory m and four components: input, generalization, output and response.
Building an intelligent dialogue agent is a long-term goal of machine learning. However, we are still far away from it. Hence a toy task, bAbi(Jason et al.,2015) is introduced, which evaluate models ability of reading comprehension via question answering. It has a 20 sub tasks, including single two or three supporting fact, yes or no question, counting, simple negation and so on. Most memory-based memories use this data set to evaluate their own performances, and to see how their models work for each sub set of this task.
End-to-end memory network(Sainbayar et al.2015) is a form of Memory Network, but can be trained end-to-end, and requires less supervision during training. It has input memory, which use inner product of input and query to get probability *p* distribution; output memory, use attention mechanism to compute *output vector*; generating final prediction by layer transformation of output and query vector.
Dynamic memory network(A Kumar, et al.,2015), introduce a episodic memory, it first use attention mechanism to focus on relevant parts, then produce a ‘memory’ vector representation by taking account of previous memory and query though a modified GRU. Multiple hops is used to do transitive inference.
Key-value memory network(A H.Miller et al., 2016) is used for reading documents. The addressing state is based on the key memory while the reading state uses the value memory. The key is designed with features to help match it to the question, while the value is designed with features to help match it to the answer.
Recurrent entity network(M Henaff et al, 2016), where our model mainly based from, it has a simple parallel architecture that several memory locations can be updated simultaneously. It sets a new state-of-the art on the bABI tasks, and is the first method to solve all the tasks in the 10k training examples setting. We will describe this model in detail soon.

3. Approach

We will first show the big picture of the model. Then discuss each component.

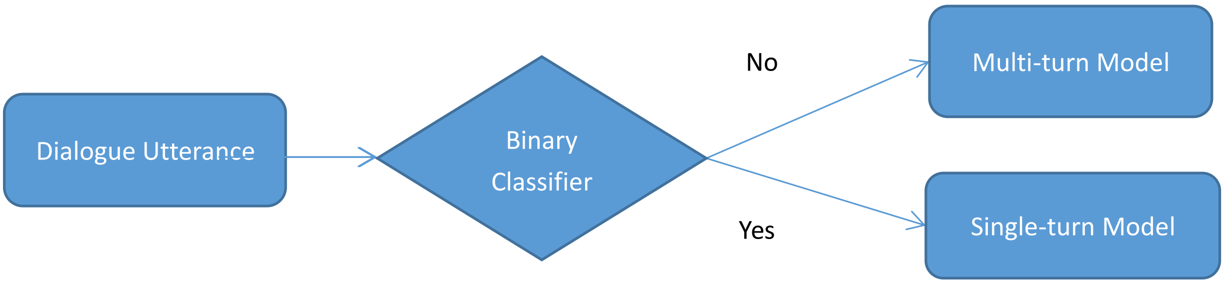

Figure 1: our architecture. Input is dialogue utterance, it will first come to a binary classifier. This classifier will decide whether it goes to multiple turn model(indicate by 'No') or single-turn model(indicate by 'Yes') based on a specific rule. This rule can be choose freely according to the need of the task. We will discuss it below.

3.1 Dialogue Utterance

Dialogue utterance usually is a sentence or sometimes multiple sentences from a user. Each token in the sentence(s) will be embedded. Then positional bag of words (bow) representation will be used to get a utterance representation. But you can also choose other commonly used sentence representation like bi-directional RNN to get the representation.

Let the input at time t be a sequence of words with embeddings {e1,...,ek}. The vector representation of this input is then:

$$S_t = \sum p_i \otimes e_i$$

The same set of vectors {p1,...,p1} are used at each time step, which represent positional information and are learn-able parameters. Element-wise product is used for position and embedding.

3.2 Binary Classifier

Take dialogue utterance representation S_t as input, binary classifier will output binary result(No,Yes). The purpose of binary classifier is to determine whether we should go to multiple turn model or single-turn model. How should we decide where to go? We have two ways:

The first method is to train a binary classifier to learn whether query can be answered without context. If a query can be answered without take context into consideration, then query is independent from context, thus no need to pass context information to our model, and single-turn model can be used; on the other hand, if query should be answered with information from context, we should pass query together with context to our model. This is very nature way to tackle multiple turn problem in dialogue. If we can have a good classifier to do this, we have already taken an important step toward solving the problem. However, this method require labeled data of whether a query is independent from context or not. See Figure 2 for some examples:

ID	Previous Query	Previous Response	Query
1	我有 5 万块怎么不可以炒创业板? I already have 50,000 balance? Why can't i trade ChiNext stock?	可能是因为您的账号没有开通创业板权限，您可以到营业部开通。 you may have no privilege access to ChiNext account, you can go to bussiness department to open it.	都有些什么条件呢? What are the conditions?
2	软件很卡 Software is very slow	请问你使用什么软件? Which software are you using?	手机软件. Mobile phone software
3	你好 Hi	您好，请问有什么可以帮到您? Hi, what can i do for you?	不记得账号了 I forgot my account.
4	怎么看开户进度 How to check status of account application?	当前 15 点前申请当天开立账号，15 电话后 T+1 天开立。会以短信告知你结果。 The same day open account if you apply before 3p.m. Otherwise, open at next day. We will inform you the result by text message.	无法购买上证股票. Unable to buy shanghai stock.

Figure 2: the first two examples show that the query should be answered by take consideration of context (ID:1,2); the later two examples show that the query can be answered without context (ID: 3,4).

The second method is train a intention recognition module. The input is dialogue utterance, the output is user's intention. If we detect two successive turn of user has different intentions, then we assume that user swift his or her intention. So we will discard context information, and only pass query to our model; otherwise, we believe context information is relevant when answering the query. An we will pass query together with context to our model. This is a coarse way to solve the problem. In many cases, when customers are trying to solve specific problems, they intentions will remain the same, this method can be employed.

The third method is to use a similarity module in single turn, and set a threshold *T* as confidence level. If similarity score is greater than *T*, then single module can be used; otherwise, use multiple model.

In our work, we tried the second and third method.

3.3 Multi-turn Model

We use Recurrent Entity Network(EntityNet) as our multiple turn model. We choose this model due to the following reason:

1. It uses blocks of keys and values, which is independent from each other. so it can be run in parallel. We think this is novel idea.
2. It models context and question together and it track state of world of memories.
3. This model is very simple in architecture. And it can achieve very good performance in toy task, which is prerequisite task for solve really world problem.

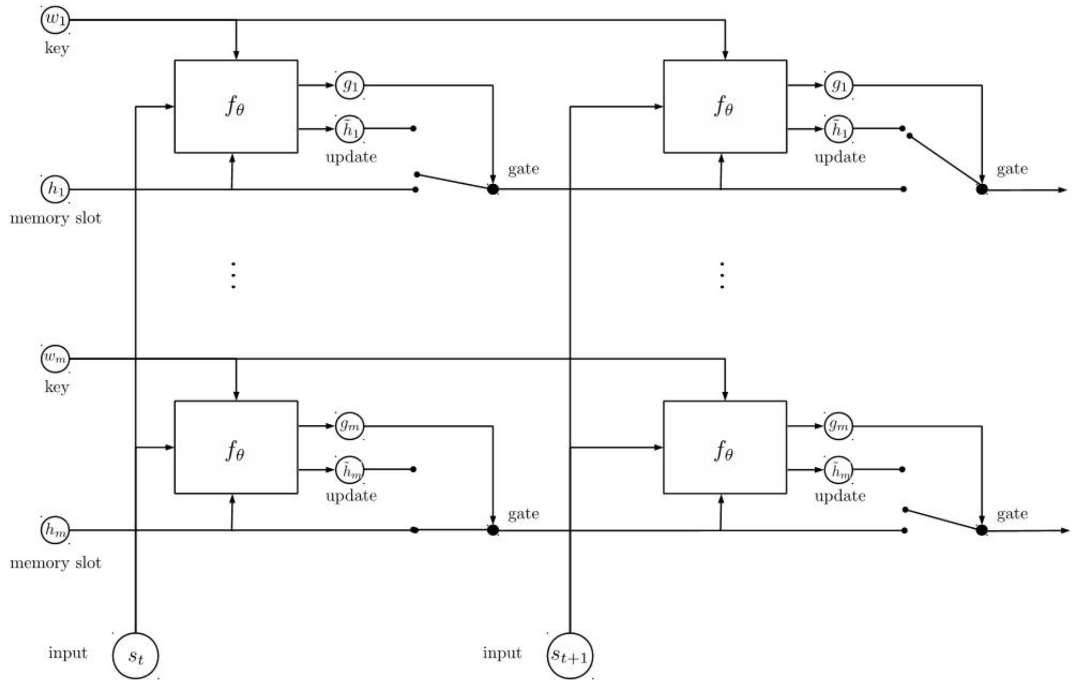

Figure 3: Diagram of dynamic memory of the Recurrent Entity Network. As you can see, there are blocks of key and memory slot run in parallel. At each time step t, new input s_t will be encoded, and *gate* function g_t will be used, then *hidden state* h_t will be updated by combine *candidate hidden state* $\tilde{h}_t$ with gate g_t .

Here we will introduce the main components of this model.

1) Input module:

It use bag of word to encode story(as context) and query(as question); it take account of position by using position mask. Multiplication is used to combine positional information and encoded vector. By using bi-directional RNN to encode story and query, performance can be boosted in some extent.

2) Dynamic memory:

It consist a series of key vectors $w_1,w_2,...w_m$ and memory(as hidden state) vectors $h_1,h_2,...h_m$. The keys represent locations, while the memories are contents. The contents(memories) get updated at inference time when new information comes.

Giving an inputs s_t, the following steps occurs:

a. Gating function. Compute gate by using similarity of keys,values with input of story.

$$g_j \leftarrow \sigma(s_t^T h_j + s_t^T w_j)$$

b. Candidate state. Get candidate hidden state by no-linearity transform key, value and input.

$$\tilde{h_j} \leftarrow \phi(Uh_j + Vw_j + Ws_t)$$

c. Update hidden state. Combine gate and candidate hidden state to update current hidden state.

$$h_j \leftarrow h_j + g_j \odot \tilde{h_j}$$

3) Output module(use attention mechanism):

a. Get possibility distribution *{p_1,p2,...,p_j}* by computing similarity of query and hidden state.

$$p_j = \text{Softmax}(q^T h_j)$$

b. Get weighted sum of hidden state *u* using possibility distribution *{p_1,p2,...,p_j}*.

$$u = \sum_j p_j h_j$$

c. Non-linearity transform of query *q* and hidden state *u* to get predicted label y.

$$y = R\phi(q + Hu)$$

3.4 Single-turn Model

A multiple label classification model is used as our single-turn model. Input is utterance from user, output is a predicted answer. We can decide whether to use a model to get predicted answer directly or use a model with similarity module to get the most relevant answer based on the similarity between the user's utterance and question. We will discuss in detail how we formulate our single-turn model in experiment section(4.3).

3.5 Value Network

In our architecture, we also design a value network module to select the most relevant answer from candidate answers that is retrieved from our policy network: multiple turn model and single turn model.

Training data as follow: {question, answer0, answer1, answer2}. Answer0 is the right answer for the question draw from dialogue history of corpus, answer 1 and answer 2 is random select from other place.

However, for some reason, such as the amount of data is not enough, or we did not construct our training data of value network in a proper way, the result from value network is is not reliable. Thus we eliminate this module from our architecture later.

4. Experiments

4.1 Data Set

We use dialogues of online customer services collected from a nationwide securities firm. It contain 1,048,576 lines with 125,314 independent dialogues. For each dialogue, it is multiple turn interact between customer service staff and customer. Average interactions between customer service staff is and customer is 7.4. Each line of a dialogue has the following columns: dialogue ID, business type, time stamp,message sender, context. It has per-defined 410 business type derived from security industry business. It has two message sender: customer, customer service staff.

Besides dialogue corpus, we also have another source of data: standard question-answer pairs. it contains 6785 question-answer pairs. The purpose of this standard question-answer pairs is to cover the most frequent questions in single turn.

Figure 4 is a sample of one dialogue; Figure 5 is two samples of question-answer pairs.

发送方	消息内容
Sender	Message Context
座席端	您好
Staff	Hello
客户端	为什么我打不了新股
Customer	Why can’t i buy new stock?
座席端	因为您没钱没有新股额度。
Staff	Because you don’t have quota of new stock.
客户端	那请问多少可以？？？额度
Customer	How much quota should i have?
客户端	之前可以打现在怎么打不了
Customer	I can’t subscribe new stock now. Previously i could.
座席端	说明您市值减少了。
Staff	It indicates that your stock’s market value is reduced.
客户端	我账户一万超了、照理应该是有额度可以打的。
Customer	My account’s value exceed 10 thousand, i should able to subscribe.
客户端	这跟我买股票没什么关系吧
Customer	It doesn’t matter. right.
座席端	有关系的。
Staff	It matters.
座席端	市值影响您的额度。
Staff	Stock market value affect quota of new stock.

Figure 4: a sample from dialogue corpus. It is a dialogue under business type of ‘new stock/rules’. Dialogue id, time stamp is omitted. For save space purpose, we choose the dialogue that sentences is not so long. But usually the response from customer services staff is much longer, could longer than 50 or 100 words. Some responses from customer services staffs may come from their working experiences, some may replicate from standard question-answer pairs. Dialogue corpus is used for multiple turn model.

问题 Question	答案 Answer
创业板权限支持多家券商吗？ It is support multiple brokers for gem trading authority?	您好，可以在多家券商开通创业板交易权限。 Hello, you can open gem trading authority in several securities brokers.
创业板权限只能在一家券商哪里开通吧 Gem trading authority can only be opened in one broker. Right?	您好，可以在多家券商开通创业板交易权限。 Hello, you can open gem trading authority in several securities brokers.

Figure 5: samples of standard question-answer pair. Generally speaking, several questions may associate with one answer. By average, it is 3 questions map to one answer. This pairs will be only used in single turn model.

4.2 Multiple Turn Model In Detail

As we discussed, we use recurrent entity network(EntityNet) as our multiple turn model. Training data is based on one million dialogue history. The percentage of training data, validation data, and test data is 80%, 10%, 10%.

There is a per-process for the dialogue corpus. We removed those dialogues that don't contain a complete interactive between staff and customer. Successive line from same sender type is concatenated so that models do not need to worry about predicting two or more utterances for a single input. If a utterance is start from customer services staff or the last utterance is ended by customer, it provide no additional information for model to predict, thus it is removed. As the dialogue is usually quite long, we split a dialogue into sub-dialogues, each dialogue contains: {context, question and response}. Context can contain multiple utterance or even no utterance. Instead of modeling last interaction of a dialogue, we try to model every interaction in the dialogue.

Concretely, for a dialogue containing $\{u_1, a_1, u_2, a_2, ..., u_{n-1}, a_{n-1}, u_n, a_n\}$, it will be split into: $\{\{\}, u_1, a_1\}, \{\{u_1, a_1\}, u_2, a_2\}, ..., \{\{..., u_{n-1}, a_{n-1}\}, u_n, a_n\}$. We can decide how long utterances included in the context for a dialogue. Although EntityNet is demonstrated has the ability to model very long input sequences, but long input sequences is computational expensive and not common in reality, so we choose modest length for the context. In our experiment, we set previous three utterances before question as our context. To get rich linguistic information and narrow the gap of huge data needed for deep learning model and modest size available data set from close domain , we used word embedding pertained from big external corpus.

4.3 Single Turn Model In Detail

In the early experiments, we employ EntityNet as our single turn model, but we set context as user's question, make it to model single question answering. Training data is the standard question answer pair. It can works in some way, but as the training data is very small, and it is prone to make wrong predictions, also lack the ability of generalization.

Later we change this single turn model. We change in the following way:

First, instead of training a model to make prediction directly, we use all of our dialogue corpus to train a intention detection model. As the same,our single turn model based on EntityNet. The input is a question, output is a business type. For example, input is a question "why can't i buy new stock?", output is "new stock/rule". As dialogue corpus and vocabulary size is much bigger than size of standard question answer pair, by training a model to predict intention detection, we can learn a better word representation, as a result, the ability of generalization is much better.

Further more, we add a similarity module to this model. For each user's question, we will compute similarity between this question and questions from standard question answer pairs using cosine similarity. Since it is implement in vectorized way, the inference speed is still very fast, and no affect to training.

Finally, we use the the top similarity score as our 'binary classifier'. When it exceed our threshold T_0, we believe it has enough confidence to retrieve the answer solely on single model. Otherwise, multiple turn model will be invoked.

By doing above changes, we improve the accuracy in a big margin, and generalization ability is improved.

4.4 Clustering: Reduce Search Space For Multiple Turn Model

Normally, Supervised Learning algorithm such as neural network needs lots of labeled data to train a model, and performance of the model due to reasons as follow:

(1). Construction of model. Aiming at specific problem, researchers required to construct the different structure of the model.

(2). Diversity of features. One of the saving graces of deep learning is it can analyze much more features than machine learning which relies on feature engineering.

(3). Complexity of target. It is obviously that the performance of a model is increasing as the number of target decreases.

As threes reasons listed above, the third one seems easier to achieve than others in some cases Therefore, unsupervised clustering algorithms were applied to our task to decrease the complexity of output space. Many classic cluster algorithms such as Affinity Propagation, DBSCAN, KMeans were analyzed, and we applied Affinity Propagation algorithm to cluster targets as it has better performance than other algorithms.

The different performances of clustering algorithms shown in Figure 6.

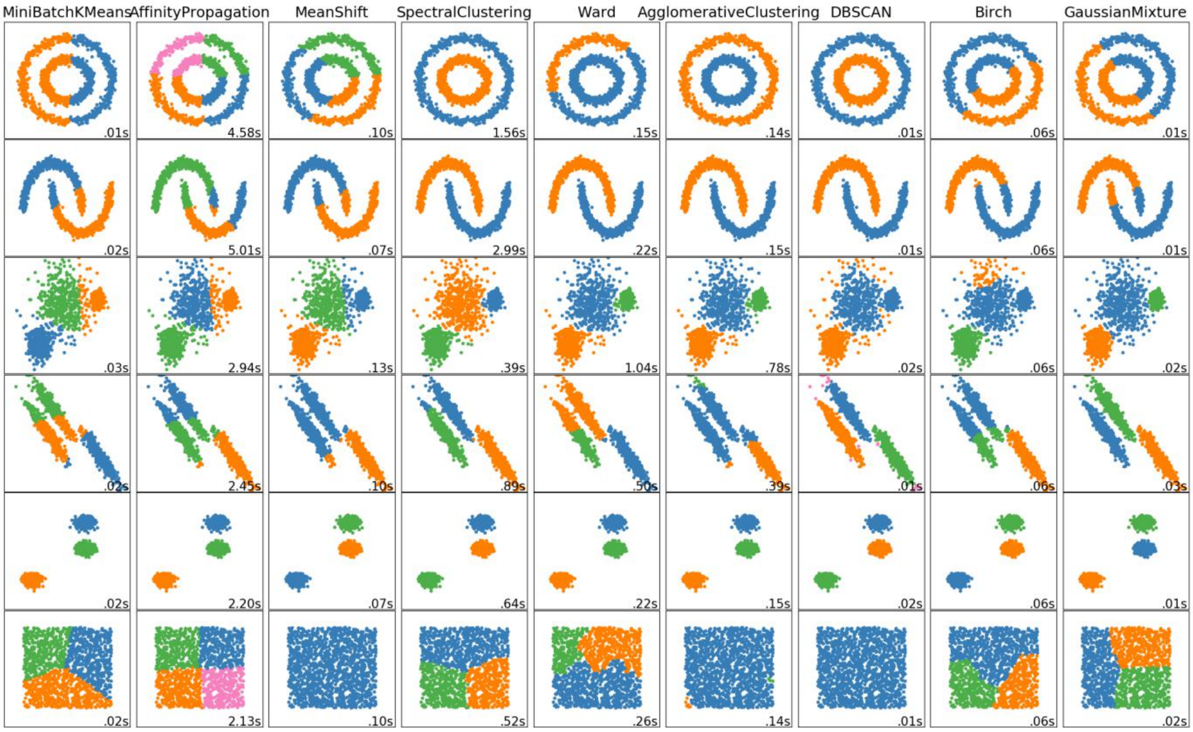

Figure 6: A comparison of the clustering algorithms from scikit-learn. It shows some methods like AffinityPropagation can find internal structure in a better way in some cases.

Affinity Propagation costs much more time than other algorithms as the kernel theory is different, in order to solve this problem we decreased the size of data set because the larger of the size the higher cost of the time. This means just a part of our data set had been clustered by Affinity Propagation, so we divided the whole data set into severals small parts, and let each part do cluster, then we calculated the center of all clusters and combined those closely clusters. The distance between each cluster center was calculated by the method of Euclidean distance.

It is mentionable that in order to improve the performance of the cluster, the corpus will be transformed into the representation of sentences embedding. The word2vector model was applied to acquire the word embedding and char embedding, then sentence embedding was combined by the layer of last hidden state from a sequence to sequence model and a positional encoding scheme proposed in (Sukhbaatar et al.2015).

{"group": 1, "sentence":
["您可以重置密码的/You can reset password",
"您可以去重置密码/You can go there to reset password",
"密码忘记可以重置/Reset password if you forget",
"那您可以重置密码/You can reset password"]}
{"group": 2, "sentence":
["您是指佣金吗？/Do you mean commission?",
"您要查佣金率吗？/Do you want to check commission rate?",
"您是要调整佣金吗？/Do you want to alter commission rate?",
"您佣金率是多少呢？/What's your commission rate?"]}

Figure 7: it presents the result of the Affinity Propagation algorithm on our corpus which included different type of questions, and it is clearly that the sentences from same type are clustered into same group.

After employee clustering technology, the total search space for multiple turn model reduced from 30,000 to 828, reduced in 2 orders of magnitude. A utterance of dialogue in original dialogue corpus which can be found in a group was replaced to a most representative sentence in this group. The representative sentences was choose by people.

4.5 Results

We use per-trained word embedding from word2vec in the training data set. Embedding size of word vector and hidden size of internal layer is set to 100. Adam is used as optimizer. L2 regularization with lambda 0.003 is used. Block of key-value pairs is set to 20. Learning rate is decay by half whenever validation loss is not decrease.

In a test done by our partner, we got a 83.3% accuracy for our hybrid model in a single turn test; for multiple turn test, we were able to show the ability to performance a continuous dialog between user and the system. Figure 8 shows two use cases of our model during inference.

User	我有 5 万块怎么不可以炒创业板? I already have 50,000 balance? Why can't i trade ChiNext stock?
Agent	可能是因为您的账号没有开通创业板权限，您可以到营业部开通。 you may have no privilege access to ChiNext account, you can go to business department to open it.
User	都有些什么条件呢？ What are the conditions?
Agent	个人投资者开通创业板交易条件:1.交易经验满两年以上;2.风险测评结果为稳健型或进取型。现场签署《风险揭示书》后开通 Individual investor can open gem trade account: 1. Two years experience of trading; 2.the risk assessment results belong to stable or aggressive. You can open account after signing the risk disclosure statement on business hall if you meet these two conditions.
User	一年不行？ How about only one year?
Agent	交易经验没满两年开通创业板原则上不可以的 General speaking it is not allow to open gem trading account if you don't have two years experience of stock trading

User	为什么我登录不了 Why can't i login in?
Agent	您是指什么？ What do you mean?
User	我登录不了你们的 app I can't login to my mobile application
Agent	错误提示是什么呢？ What's the error message?
User	提示说身份证过期了 It says ID is expired
Agent	电脑官网-网上营业厅-身份证信息修改；手机 app-掌厅-身份信息修改 Our webiste->-business hall for webiste-->ID information modification; Mobile app-->business hall for mobile-->ID information modification

Figure 8: two samples of continuous dialog between user and agent. The first example is talking about criteria for opening gem trading account; the second example is user is having trouble to login his or her app.

5. Discussion and Conclusion

Question answering is one of the key problems in natural language problems. Many problems can be cast into question answering problem. Online customer service task is a form of question answering problem. We discussed our approach, a memory-based hybrid architecture to solve this problem. Through many memory-based models demonstrated promising results on synthetic tasks, previously we are not clearly whether these models can be used in really world problems and yield good result. From the experiments, the ability of these model is further confirmed. By fusing multiple turn model and single turn model together and use unsupervised learning like cluster to reduce search space, the result in online customer service is satisfactory.

Although our approach clearly demonstrated the ability to conduct successive interaction between user and agent, there is still a long way to go in solving successive interaction in dialogue. As deep learning models need huge amount of training data to yield a good result and there is always limited corpus in close domain, in order to across the gap, more data need to be collected and new approach still need to be explored.

Reference

Hochreiter, Sepp and Schmidhuber, J¨urgen. Long short-term memory. Neural Comput., 9(8):1735– 1780,November1997. ISSN0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL *http://dx. doi.org/10.1162/neco.1997.9.8.1735.*

Kingma, Diederik P. and Ba, Jimmy. Adam: A method for stochastic optimization. CoRR, abs/1412.6980, 2014. URL *http://arxiv.org/abs/1412.6980.*

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Advances in neural information processing systems, pages 3104 – 3112, 2014. URL *https://arxiv.org/abs/1409.3215*

Weston, Jason, Chopra, Sumit, and Bordes, Antoine. “Memory networks”. CoRR, abs/1410.3916, 2014.URL *http://arxiv.org/abs/1410.3916*

Jason Weston, Antoine Bordes, Sumit Chopra, Tomas Mikolov, Alexander M. Rush, Bart van Merrenboer, "Towards AI-Complete Question Answering: A Set of Prerequisite Toy Tasks", *arXiv:1502.05698 [cs.AI]. URL https://arxiv.org/abs/1502.05698*

Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, Rob Fergus, "End-To-End Memory Networks", *arXiv:1503.08895 URL https://arxiv.org/abs/1503.08895*

Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, Richard Socher.”Ask Me Anything: Dynamic Memory Networks for Natural Language Processing”, *arXiv:1506.07285 URL https://arxiv.org/abs/1506.07285*

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, Jason Weston, “Key-Value Memory Networks for Directly Reading Documents”, arXiv:1606.03126 URL https://arxiv.org/abs/1606.03126

Mikael Henaff, Jason Weston, Arthur Szlam, Antoine Bordes, Yann LeCun,”Tracking the World State with Recurrent Entity Networks”,arXiv: 1612.03969 URL https://arxiv.org/abs/1612.03969

Clustering — scikit-learn documentation URL http://scikit-learn.org/stable/modules/clustering.html#clustering