

METHODS AND SYSTEMS FOR RECOMMENDATION USING A NEURAL NETWORK

TECHNICAL FIELD

The present disclosure relates to recommendation systems and methods such as systems and methods for making a culinary recommendation using a neural network.

BACKGROUND

Recommendation systems often provide recommendations from large numbers of recommendation options, for example in culinary recommendation systems, a recommendation of merchants from a large number of merchants may be provided. Neural networks provide a trainable method of recommendation that can take into account a variety of factors or features. However, as the number of recommendation options increases, the amount of processing required to provide recommendations also increases. This hinders the provision of real time recommendations.

SUMMARY

According to a first aspect of the present disclosure, a method of generating a recommendation for a target user based on an input using a neural network is provided. The input comprises an indication of the target user. The method comprises: identifying a target user cluster to which the target user is assigned from a plurality of user clusters; looking up in a stored table, cluster scores for a plurality of candidate recommendation options for the target user cluster, the stored table comprising cluster scores for each recommendation option of the plurality of recommendation options; ranking the plurality of candidate recommendation options according to the cluster scores for the target user cluster; selecting a target set of recommendation options from the plurality of candidate recommendation options according to the ranking of the plurality of candidate recommendation options; calculating a set of user specific recommendation scores corresponding to the target set of recommendation options using the neural network; ranking the target set of recommendation options using the user specific recommendation scores; and

generating a recommendation for the target user based on the ranking of the target set of recommendation options.

The input may further comprise an indication of a location of the target user, and each recommendation option may be associated with a location. Selecting a target set of recommendation options from the plurality of candidate recommendation options may then comprise selecting the target set of recommendation options from recommendation options associated with a location within a threshold distance of the location of the target user.

The recommendation options may be, for example, merchants which may be culinary merchants such as restaurants or takeaways.

The methods allow real-time or near real-time processing to generate recommendation options. One reason for this is that the cluster scores can be pre-calculated in an offline pre-training method for a large number of recommendation options. Then when the real-time or near real time online processing is carried out, processing using the neural network is carried out only for plurality of candidate recommendation options.

The method may comprise calculating a set of user specific recommendation scores corresponding to the set of candidate recommendation options using the neural network by inputting contextual features into the neural network.

The input may further comprise an indication of the user location and each recommendation option is associated with a location and wherein the contextual features comprise a distance between the location of the user and the location associated with a respective recommendation option.

The input may further comprise an indication of a time stamp and wherein the contextual features comprise a feature dependent on the timestamp. This allows the method to take into account factors such as the time of day. It is noted that for culinary recommendations this may be an important factor since different restaurants

or merchant may be popular with customers at different times of day corresponding to different meal times.

The recommendation for the target user may comprise a ranked list of recommendation options.

The neural network may comprise a user subnetwork that maps user features onto a user embedding space and the target user cluster corresponds to a user cluster in the user embedding space.

According to a second aspect of the present disclosure, a method of generating a recommendation for a target user based on an input using a neural network is provided. The input comprises an indication of the target user. The method comprises: selecting a target set of recommendation options from a plurality of candidate recommendation options; calculating a set of user specific recommendation scores corresponding to the target set of recommendation options using the neural network; ranking the target set of recommendation options using the user specific recommendations scores; and generating a recommendation for the target user based on the ranking of the target set of recommendation options. Such embodiments provide a modular recommendation system which allows the creation, testing and implementation of new recommendation modules. Different possible methods of generating the target set of recommendation options may be implemented in such embodiments.

According to a third aspect of the present disclosure a pre-training method in a recommendation system is provided. The recommendation system comprises data storage for a neural network configured to generate recommendation scores for recommendation options. The neural network comprises a user subnetwork that maps user features onto a user embedding space. The pre-training method comprises: training the neural network; clustering users in the user embedding space using the user subnetwork of the trained neural network to generate embedding space user clusters; calculating cluster scores for the embedding space user clusters for each of a plurality of recommendation options; and storing the calculated cluster scores and the trained neural network in the data storage.

The cluster scores may be calculated by calculating cluster scores for the embedding space user clusters for each of a plurality of recommendation options comprises determining a cluster center embedding vector in the embedding space for each embedding space user cluster and calculating the cluster scores using the cluster center embedding vectors.

In embodiments, the clustering methodology differs from traditional approaches. Typically, in traditional approaches, one would cluster users based on user features such as demographics and interaction behaviors. However, as these data points are often noisy and difficult to obtain, in the present disclosure, we instead cluster a user representation that is learnt as part of the neural network training. As such, the clusters obtained are more representative of how a user would react towards the target variable.

According to a fourth aspect of the present disclosure a recommendation system for generating a recommendation for a target user based on an input is provided. The input comprises an indication of the target user. The recommendation system comprises a processor, data storage, and program storage. The data storage stores a neural network and stored table. The neural network is configured to generate recommendation scores for recommendation options. The stored table comprises cluster scores for each recommendation option of a plurality of recommendation options. The program storage stores computer program instructions operative by the processor to; look up cluster scores for a plurality of candidate recommendation options for a target user cluster in the stored table; rank the plurality of candidate recommendation options according to the cluster scores for the target user cluster; select a target set of recommendation options from the plurality of candidate recommendation options according to the ranking of the plurality of candidate recommendation options; calculate a set of user specific recommendation scores corresponding to the target set of recommendation options using the neural network; rank the target set of recommendation options using the user specific recommendations scores; and generate a recommendation for the target user based on the ranking of the target set of recommendation options.

According to a fifth aspect of the present disclosure, a recommendation system for generating a recommendation for a target user based on an input is provided. The input comprises an indication of the target user. The recommendation system comprises a processor, data storage, and program storage, the data storage storing a neural network, the neural network configured to generate recommendation scores for recommendation options. The program storage storing computer program instructions operative by the processor to; select a target set of recommendation options from a plurality of candidate recommendation options; calculate a set of user specific recommendation scores corresponding to the target set of recommendation options using the neural network; rank the target set of recommendation options using the user specific recommendations scores; and generate a recommendation for the target user based on the ranking of the target set of recommendation options.

According to a further aspect of the present disclosure a computer readable medium carrying computer executable instructions which when executed on a processor cause the processor to carry out a method according to one of the above aspects is provided.

BRIEF DESCRIPTION OF THE DRAWINGS

In the following, embodiments of the present invention will be described as non-limiting examples with reference to the accompanying drawings in which:

FIG.1 is a block diagram showing the message flow between a recommendation system according to an embodiment of the present invention and a user device;

FIG.2 is a block diagram showing recommendation system according to an embodiment of the present invention;

FIG. 3 is a block diagram showing neural network according to an embodiment of the present invention;

FIG.4 is a flow chart showing a method of pre-training according to an embodiment of the present invention;

FIG.5 illustrates clustering of customers in an embedding space carried out in a method according to an embodiment of the present invention;

FIG.6 is a table showing calculated cluster scores for merchants according to an embodiment of the present invention; and

FIG.7 is a flowchart showing a culinary recommendation method using a neural network according to an embodiment of the present invention.

DETAILED DESCRIPTION

The present disclosure relates recommendation systems, and for example, to culinary recommendation, that is the recommendation of culinary or food and beverage merchant such as a restaurant, take-away, bar, or similar establishment, or the recommendation of dishes at a culinary merchant.

FIG.1 is a block diagram showing the message flow between a recommendation system according to an embodiment of the present invention and a user device. As shown in FIG.1, user device 150 associated with a customer or user generates a recommendation request 160, which is sent to the recommendation system 100. The recommendation request 160 comprises an indication of a user identifier 162, an indication of a user location 164 and an indication of a time stamp of the current time 166 at which the recommendation 160 request was generated. The recommendation request 160 may be generated by an application running on the user device 150, for example in response to the user of the user device 150 requesting a culinary recommendation. The recommendation system 100 may provide the functionality of a culinary recommendation platform that allows the customer or user of the user device 150 to make an order at a take-away food merchant or book a table at a restaurant.

In response to receiving the recommendation request 100, the recommendation system 100 generates a recommendation response 170. The recommendation

response 170 comprises a ranked list of recommendations 172. The ranked list of recommendations 172 may comprise a ranked list of recommended dishes for the customer or a ranked list of culinary merchants.

FIG.2 is a block diagram showing recommendation system according to an embodiment of the present invention.

The recommendation system 100 comprises a processor 110, a working memory 112, a network interface 114, program storage 120 and data storage 130. The processor 110 may be implemented as one or more central processing unit (CPU) chips. The program storage 120 is a non-volatile storage device such as a hard disk drive which stores computer program modules. The computer program modules are loaded into the working memory 112 for execution by the processor 110. The network interface 114 is an interface that allows the recommendation system 100 to communicate with devices such as the user device 150 shown in FIG.1.

The program storage 120 stores a neural network training module 122, a clustering module 124, a cluster score module 126 and a recommendation module 128. The computer program modules cause the processor 110 to execute various recommendation and pre-training methods which are described in more detail below. The program storage 120 may be referred to in some contexts as computer readable storage media and/or non-transitory computer readable media.

In addition to the aforementioned modules, the program storage 120 may also store an evaluation module which calculates evaluation metrics (e.g. area under the receiver operating characteristic curve) for each cluster, then aggregates those cluster metrics to summarize the performance of the neural network. This also allows the performance of quality checks to prevent the deployment of models if the performance of one cluster is extremely poor. In contrast to existing evaluation procedures, this implementation allows the performance of specific sub-segments of the dataset to be evaluated.

As depicted in FIG.2, the computer program modules are distinct modules which perform respective functions implemented by the recommendation system 100. It

will be appreciated that the boundaries between these modules are exemplary only, and that alternative embodiments may merge modules or impose an alternative decomposition of functionality of modules. For example, the modules discussed herein may be decomposed into sub-modules to be executed as multiple computer processes, and, optionally, on multiple computers. Moreover, alternative embodiments may combine multiple instances of a particular module or sub-module. It will also be appreciated that, while a software implementation of the computer program modules is described herein, these may alternatively be implemented as one or more hardware modules (such as field-programmable gate array(s) or application-specific integrated circuit(s)) comprising circuitry which implements equivalent functionality to that implemented in software.

Although the recommendation system 100 is described with reference to a computer, it should be appreciated that the recommendation system 100 may be formed by two or more computers in communication with each other that collaborate to perform a task. For example, but not by way of limitation, an application may be partitioned in such a way as to permit concurrent and/or parallel processing of the instructions of the application. Alternatively, the data processed by the application may be partitioned in such a way as to permit concurrent and/or parallel processing of different portions of a data set by the two or more computers. In an embodiment, virtualization software may be employed by the recommendation system 100 to provide the functionality of a number of servers that is not directly bound to the number of computers in the recommendation system 100. In an embodiment, the functionality disclosed above may be provided by executing the application and/or applications in a cloud computing environment. Cloud computing may comprise providing computing services via a network connection using dynamically scalable computing resources. A cloud computing environment may be established by an enterprise and/or may be hired on an as-needed basis from a third party provider.

The data storage 130 stores a neural network 132 and a cluster score table 134 examples of which are described below with reference to FIG.3 and FIG.6 respectively.

The recommendation system 100 is coupled to a set of databases which store merchant and / or dish metadata 140, merchant and / or dish interaction data 142, customer metadata 144, customer interaction data 146 and historical purchase and / or booking data 148. The merchant and / or dish metadata 140 may comprise indications of cuisine, price and / or ratings. The merchant and / or dish interaction data 142 may comprise an indication of the number of times that a merchant has been favorited, number of purchases for a specific mealtime for a merchant / dish. The customer metadata 144 may comprise indications of the occupation, and age on the platform of the user. The customer interaction data 146 may comprise an indication of the time a user has spent using the platform, and / or the frequency and timing of the user's interactions with the platform. The historical booking data may comprise purchase transaction logs for the user.

FIG. 3 is a block diagram showing neural network according to an embodiment of the present invention. The neural network 300 shown in FIG.3 is an example of the neural network 132 stored in the data storage 130 of the recommendation system 100 shown in FIG.2.

The neural network 300 shown in FIG.3 is configured for generation of merchant recommendations, however, it will be appreciated by those of skill in the art that the neural network 300 may be adapted to other types of recommendation, for example the recommendation of culinary dishes.

The neural network 300 comprises a merchant subnetwork 310, a customer subnetwork 330 and a main recommendation network 350. The merchant subnetwork 310 generates merchant embeddings 320 from merchant metadata features 312 and merchant interaction features 314. The customer subnetwork 330 generates customer embeddings 340 from customer metadata features 332 and customer interaction features 334. The merchant embeddings 320, the customer embeddings 340 and contextual features 360 are inputs to the main recommendation network 350. The main recommendation network 350 generates a prediction probability 356 in response to the inputs.

The customer interaction features may involve a sequence of events (hence, the 3 layers as depicted). For example, this may be a representation of the last 3 merchants that a user has purchased from. We make the implicit assumption here that meaningful user representations can be obtained by aggregating their respective merchant representations.

The prediction probability 356 represents a probability that a customer having the customer metadata features 332 and the customer interaction features 334 will select a merchant having the merchant metadata features 312 having interaction features 314 when the contextual features 360 are present. The contextual features 360 represent variables such as the time of day, the day of the week, the weather conditions, traffic conditions, and other variables. In some embodiments, the contextual features may comprise a distance between the location of the customer and the location of the merchant.

The merchant subnetwork 310 comprises a merchant metadata features input layer 316 which takes the merchant metadata features 312 as an input and combines the input merchant metadata features 312 according to weights to generate an output function. The weights are determined during a training process. The merchant subnetwork 310 also comprises a merchant interaction feature input layer 318 which takes the merchant metadata features 314 as an input and combines the input merchant metadata features 314 according to weights to generate an output function. The weights are determined during a training process. The merchant metadata features input layer 316 and the merchant interaction feature input layer 318 are coupled to an output layer that generates the merchant embeddings 320 which are projections of the merchant metadata features 312 and the merchant interaction features 314 onto an embedding space.

The customer subnetwork 330 comprises a customer metadata features input layer 336 which takes the customer metadata features 332 as input and combines the customer metadata features 332 according to weights to generate an output function. The weights are determined during a training process. The customer subnetwork 330 further comprises a customer interaction features input layer 338 which takes the customer interaction features 334 as input and combines the customer

interaction features 334 according to weights to generate an output. The weights are determined during a training process. The customer metadata features input layer 336 and the customer interaction features input layer 338 are coupled to an output layer that generates the customer embeddings 340 which are projections of the customer metadata features 332 and the customer interaction features 334 onto an embedding space.

The main recommendation network 350 comprises an input layer 352 which takes the merchant embeddings 320, the customer embeddings 340 and the contextual features 360 as input, and an output layer 354 which generates the prediction probability 354 as an output.

It should be noted that while FIG.3 shows each of the merchant subnetwork 310, the customer subnetwork 330 and the main recommendation network 350 as having an input layer and an output layer, they may additionally comprise one or more intermediate or hidden layers. Each of the layers of the merchant subnetwork 310, the customer subnetwork 330 and the main recommendation network 350 comprise a plurality of nodes which are linked to nodes in neighboring layers by edges. The process of training the neural network 300 involves optimizing the weights associated with the edges.

In order to allow real-time or near real-time recommendation using the neural network 300 shown in FIG.3, an offline pre-training method is carried out. In the offline pre-training method, the neural network 300 is trained using historic purchase or booking data, then clusters of customers in the customer embedding space are determined. These clusters are used to generate cluster scores for each of the merchants. In the online processing, real-time or near real-time recommendation method, the cluster scores are used to identify a candidate set of merchants for recommendation. Then, the neural network 300 is used to determine prediction probability scores for each of the candidate set of merchants and the merchants having the highest prediction probability scores are selected for recommendation.

FIG.4 is a flow chart showing a method of pre-training according to an embodiment of the present invention. The method 400 shown in FIG.4 is carried out by the recommendation system 100 shown in FIG.2.

In step 402, the neural network training module 122 is executed by the processor 110 of the recommendation system 100 and the neural network 300 is trained by the neural network training module 122 using the historical booking and / or purchase data 148. The historical booking and / or purchase data 148 shows purchases or bookings made by customers at merchants. During the training of the neural network 300, the weights of the layers are optimized to minimize a difference function between the observed bookings at merchants by customers and the prediction probability 356 output by the neural network 300 when the inputs are the merchant metadata, merchant interaction data, customer metadata, and customer interaction data for the customer and merchant. In addition, the historical booking and / or purchase data 148 may also comprise contextual data such as the time of the booking or purchase, weather and traffic conditions on the day of the booking or purchase and a distance between the customer's location and the merchant location at the time of the booking or purchase.

The difference function which is minimized in step 402 may be with respect to cross-entropy; this describes the ability of the neural network to learn the probability distribution of the training dataset. In this case, the training set is a set of positive examples (bookings) and negative examples (clicked, but did not book).

Once the neural network 300 is trained, the trained neural network is stored in the data storage 130 of the recommendation system 100 as neural network 132.

In step 404, the cluster score module 126 is executed by the processor 110 of the recommendation system 100. The cluster score module 126 generates clusters of customers. Customers are clustered in the embedding space of the customer embeddings 340. An example of clustering of customers is shown in FIG.5

FIG.5 illustrates clustering of customers in an embedding space carried out in a method according to an embodiment of the present invention. FIG.5 shows the

customer embedding space 500 of the customer embeddings 340. The customers 502 are shown as dots in the embedding space 500. As shown in FIG.5, the customers 502 are grouped into clusters. FIG.5 shows cluster a 504a, cluster b, 504b, cluster c 504c, cluster d, 504d and cluster m 504m. It is noted that while FIG.5 depicts a 2-dimensional embedding space, it is envisaged that the actual space may be multidimensional (order of magnitude in the hundreds).

A cluster center embedding vector is calculated for each cluster. As shown in FIG.5, cluster a has a cluster center embedding vector 506a, cluster b has a cluster center embedding vector 506b, cluster c has a cluster center embedding vector 506c, cluster d has a cluster center embedding vector 506d and cluster m has a cluster center embedding vector 506m.

Returning now to FIG.4, in step 506, the cluster score module 126 stored in the program storage 120 of the recommendation system 100 is executed by the processor 110. The cluster score module 126 calculates cluster scores using the neural network 300 which corresponds to the neural network 132 stored in the data storage 130. In order to calculate the cluster scores, the cluster center embedding vectors 506a, 506b, 506c, 506d and 506m are used as an input to the main recommendation network 350, the merchant metadata features 312 and merchant interaction features 314 for each merchant are also input into the merchant subnetwork 310. Using the output merchant embeddings 310 and the cluster center embedding vectors 506a, 506b, 506c, 506d and 506m, cluster scores are calculated for each of the merchants. It is noted that in the calculation of the cluster scores, contextual features 360 are not input into the main recommendation network 350.

In step 508, the cluster score module 126 stored in the program storage 120 of the recommendation system 100 is executed by the processor 110 to store the calculated cluster scores in the cluster score table 134 of the data storage 130. An example of the cluster score table 134 is shown in FIG.6.

FIG.6 is a table showing calculated cluster scores for merchants according to an embodiment of the present invention. As shown in FIG.6, the table 600 comprises a row for each merchant. Columns of the table correspond to each of the clusters of

customers. As shown in FIG.6, merchant n has cluster scores $a(n)$, $b(n)$, $c(n)$, $d(n)$... $m(n)$ for each of the customer clusters.

The method shown in FIG.4 may be repeated at regular intervals, for example on a daily basis. The pre-training carried out in the method shown in FIG.4 allows the processing requirements of real-time or near real-time online recommendation to be reduced. The online recommendation processing is described below with reference to FIG.7.

FIG.7 is a flowchart showing a culinary recommendation method using a neural network according to an embodiment of the present invention. The method 700 shown in FIG.7 is carried out by the recommendation system 100 shown in FIG.2.

In step 702, the recommendation system 100 receives a recommendation request. The recommendation request may be received by the network interface 114 of the recommendation system 100. As shown in FIG.1, the recommendation request 160 may comprise an indication of a user identifier 162, an indication of a user location 164 and an indication of the current time 166.

In step 704, the recommendation module 128 is executed by the processor 110 of the recommendation system 100 to identify a target user cluster. As described above with reference to FIG.5, the customers are clustered in the customer embedding space. As a result of this clustering, each customer is assigned to a cluster. The data storage 130 may store indications of which cluster each customer is assigned to. These indications may comprise a table or mapping of user identifiers to clusters. Step 704 may comprise using the table or mapping to determine the target cluster to which the user corresponding to the user identifier 162 indicated in the recommendation request 160 is assigned.

In step 706, the recommendation module 128 executed by the processor 110 of the recommendation system 100 looks up merchant cluster scores for the target user cluster. Step 704 comprises accessing the cluster score table 134 stored in the data storage 130 of the recommendation system 100.

In step 708, the recommendation module 128 executed by the processor 110 of the recommendation system 100 ranks the merchants according to the cluster scores for the target user cluster.

In step 710, the recommendation module 128 executed by the processor 110 of the recommendation system 100 selects a target set of merchants according to the cluster score ranking determined in step 708. Step 710 may comprise selecting a set number of the highest ranked merchants, for example the top 200 ranked merchants for the target user cluster. In some embodiments, the target set of merchants may be identified using a distance between the merchant location and the user location. For example, only merchants within a threshold distance of the user may be included in the ranking. The threshold distance may be, for example, 2km.

In step 712, the recommendation module 128 executed by the processor 110 of the recommendation system 100 calculates target user scores for the target set of merchants. Step 712 comprises using the neural network 300 to calculate the target user scores for each merchant of the target set of merchants. The inputs to the neural network 300 are, for the merchant under consideration, the merchant metadata features 312 and the merchant interaction features 314, the customer metadata features 332 for the user, the customer interaction features for the user and contextual features 360 corresponding to the time indicated by the indication of the current time 166. In some embodiments, the contextual features may also comprise a distance between the location of the user indicated by the user location 164 and the location of the merchant which may be determined from the merchant metadata. For each merchant, a target user score is calculated as the prediction probability 356 output by the neural network 300.

In step 714, the recommendation module 128 executed by the processor 110 of the recommendation system 100 to rank the merchants of the target set of merchants according to the target user scores calculated in step 712.

In step 716 the recommendation module 128 executed by the processor 110 of the recommendation system 100 generate a merchant recommendation according to

the ranking determined in step 714. The merchant recommendation may comprise a ranked list of merchants or may comprise an indication of a single merchant.

Following step 716, the merchant recommendation is sent to the user device 150 as the recommendation response 172 as shown in FIG.1. In some scenarios, there may be an additional step following step 716 prior to sending the merchant recommendation to the user device. This step may comprise additional constraints such as ensuring that the ranked list of merchants does not include multiple merchants of the same brand (for example multiple merchants having the same brand but at different locations. This additional step may also involve filtering by distance from the user.

As mentioned above, in some embodiment, the distance between the target user and the merchant is used as a contextual feature input to the neural network 300. In an alternative embodiment, the distance is used to filter results from the neural network.

In some embodiments, the system is also built in a modular fashion, which allows to implementation and testing of new recommendation modules. For example, steps 704 706 708 and 710 shown in FIG.7 may be replaced with an alternative method to select target merchants. Then recommendations may be generated using the alternative method which may not involve the calculation of cluster scores.

Whilst the foregoing description has described exemplary embodiments, it will be understood by those skilled in the art that many variations of the embodiments can be made within the scope and spirit of the present invention.

CLAIMS

1. A method of generating a recommendation for a target user based on an input using a neural network, the input comprising an indication of the target user, the method comprising:
 - identifying a target user cluster to which the target user is assigned from a plurality of user clusters;
 - looking up in a stored table, cluster scores for a plurality of candidate recommendation options for the target user cluster, the stored table comprising cluster scores for each recommendation option of the plurality of recommendation options;
 - ranking the plurality of candidate recommendation options according to the cluster scores for the target user cluster;
 - selecting a target set of recommendation options from the plurality of candidate recommendation options according to the ranking of the plurality of candidate recommendation options;
 - calculating a set of user specific recommendation scores corresponding to the target set of recommendation options using the neural network;
 - ranking the target set of recommendation options using the user specific recommendations scores; and
 - generating a recommendation for the target user based on the ranking of the target set of recommendation options.
2. A method according to claim 1, wherein the input further comprises an indication of a location of the target user, and each recommendation option is associated with a location and wherein selecting a target set of recommendation options from the plurality of candidate recommendation options comprises selecting the target set of recommendation options from recommendation options associated with a location within a threshold distance of the location of the target user.
3. A method according to claim 1 or claim 2, wherein the recommendation options are merchants.

4. A method according to any preceding claim wherein calculating a set of user specific recommendation scores corresponding to the set of candidate recommendation options using the neural network comprises inputting contextual features into the neural network.
5. A method according to claim 4 wherein the input further comprises an indication of a time stamp and wherein the contextual features comprise a feature dependent on the timestamp.
6. A method according to any preceding claim wherein the recommendation for the target user comprises a ranked list of recommendation options.
7. A computer readable medium carrying computer executable instructions which when executed on a processor cause the processor to carry out a method according to any one of the preceding claims.
8. A recommendation system for generating a recommendation for a target user based on an input, the input comprising an indication of the target user, the recommendation system comprising a processor, data storage, and program storage, the data storage storing a neural network and stored table, the neural network configured to generate recommendation scores for recommendation options, the stored table comprising cluster scores for each recommendation option of a plurality of recommendation options, the program storage storing computer program instructions operative by the processor to;
 - look up cluster scores for a plurality of candidate recommendation options for a target user cluster in the stored table;
 - rank the plurality of candidate recommendation options according to the cluster scores for the target user cluster;
 - select a target set of recommendation options from the plurality of candidate recommendation options according to the ranking of the plurality of candidate recommendation options;
 - calculate a set of user specific recommendation scores corresponding to the target set of recommendation options using the neural network;

rank the target set of recommendation options using the user specific recommendations scores; and
generate a recommendation for the target user based on the ranking of the target set of recommendation options.

9. A recommendation system according to claim 8 wherein the input further comprises an indication of a location of the target user, and each recommendation option is associated with a location and wherein the program storage stores computer program instructions operative by the processor to: select the target set of recommendation options from recommendation options associated with a location within a threshold distance of the location of the target user.

10. A recommendation system according to claim 8 or claim 9, wherein the recommendation options are merchants.

11. A recommendation system according to any one of claims 8 to 10 wherein the program storage stores computer program instructions operative by the processor to: calculating the set of user specific recommendation scores corresponding to the set of candidate recommendation options using the neural network by inputting contextual features into the neural network.

12. A recommendation system according to claim 11 wherein the input further comprises an indication of a time stamp and wherein the contextual features comprise a feature dependent on the timestamp.

13. A recommendation system according to any one of claims 8 to 12 wherein the recommendation for the target user comprises a ranked list of recommendation options.

14. A recommendation system according to any one of claims 8 to 13 wherein the neural network comprises a user subnetwork that maps user features onto a user embedding space and wherein the target user cluster corresponds to a user cluster in the user embedding space.

15. A recommendation system according to any one of claims 8 to 14 wherein the program storage stores computer program instructions operative by the processor to:

- training the neural network;

- cluster users in the user embedding space using the user subnetwork of the trained neural network to generate embedding space user clusters;

- calculate cluster scores for the embedding space user clusters for each of a plurality of recommendation options; and

- store the calculated cluster scores and the trained neural network in the data storage.

16. A recommendation system according to claim 15, wherein the program storage stores computer program instructions operative by the processor to: calculate the cluster scores for the embedding space user clusters for each of a plurality of recommendation options by determining a cluster center embedding vector in the embedding space for each embedding space user cluster and calculating the cluster scores using the cluster center embedding vectors.