A Heterogeneous Conversational Recommender System for Financial Products

Mao Kang
Ping An Technology (Shenzhen) Co., Ltd
Shanghai, China
kangmao028@pingan.com.cn

Ye Bi
Ping An Technology (Shenzhen) Co., Ltd
Shanghai, China
biye645@pingan.com.cn

Zhenyu Wu
Ping An Technology (Shenzhen) Co., Ltd
Shanghai, China
wuzhenyu447@pingan.com.cn

Jianming Wang
Ping An Technology (Shenzhen) Co., Ltd
Shenzhen, China
wangjianming888@pingan.com.cn

Jing Xiao
Ping An Technology (Shenzhen) Co., Ltd
Shenzhen, China
xiaojing661@pingan.com.cn

ABSTRACT

Financial products recommendation distinguishes itself from e-commerce and web recommendation. Financial products have fewer available items, are more expensive, less frequently purchased and subject to user specific constraints. The study in financial products recommendation is quite limited and current industry application is still focusing on exploiting machine learning techniques. Behavioral Finance theory states financial decisions are affected by psychological behavior biases, which are generally identified via conversation with professional advisors. Besides, in a conversation customer actively express subjective requirements and interests, which cannot be known from their static structured data. Inspired by that, we propose an innovative heterogeneous conversational recommender system (HConvoNet) which will consider not only customer's static profile but also the implicit behavior biases and interests, thus is adaptive to customer. The proposed framework consists of two modules: profile module and conversation module. The profile module aims to capture customer's important static needs, while the conversation module aims to extract behavior biases and dynamic interests. By integrating profile module and conversation module, HConvoNet can recommend financial products in an adaptive way. The experiments are conducted on three internal datasets from Ping An Insurance and try to predict customer's purchase intention. We compare our model with several baselines and see that our proposed model has a significant improvement.

CCS CONCEPTS

• **Information systems** → **Recommender systems**; • **Computing methodologies** → *Information extraction*;

KEYWORDS

Conversational Recommender System, Financial Products Recommendation, Heterogeneous Modelling, Deep Neural Networks

ACM Reference Format:
Mao Kang, Ye Bi, Zhenyu Wu, Jianming Wang, and Jing Xiao. 2020. A Heterogeneous Conversational Recommender System for Financial Products. In *Proceedings of KaRS2019 Second Workshop on Knowledge-Aware and Conversational Recommender Systems (KaRS 2019).* ACM, New York, NY, USA, 5 pages.

KaRS 2019, November 3rd-7th, 2019, Beijing, China.

1 INTRODUCTION

Recommender Systems are extensively used in various areas. Most of the researches focus on collaborative-filtering and content-based filtering. Collaborative-filtering assumes that users agreed in the past will like similar items. Content-based filtering tries to recommend similar items the user liked in the past. E-commerce companies like Amazon, ebay and Alibaba use well-developped collaborative-filtering algorithms to recommend products. Video and music websites like Youtube and Spotify use content-based filtering to recommend playlists.

In recent years, the research has extended to recommend financial products and insurances (we will call them together as "financial products"). Financial products recommendation is quite different from the above mentioned recommendation. E-commerce companies usually have large amount of data and frequent user actions. While, financial products have fewer available items and are not frequently purchased. Besides, they are usually more expensive and subject to user specific constraints. Knowledge-based Recommender System is a specific type of Recommender System which uses knowledge base and user profile to make personalized recommendation. It is typically applied in the domains where collaborative-filtering and content-based filtering cannot be applied, such as financial products recommendation. Most of the current studies on this topic are still in the scope of constraint-based or case-based reasoning. In practice, building knowledge base is complicated and costly, thus the practical application still relies on exploiting customer profiles using machine learning techniques for the simplicity, robustness and good explanation, such as Random Forest and Generalized Linear Models.

Recommending financial products requires thorough understanding about financial decision-making process. Behavioral Finance[15] studies the psychology of financial decision-making process. It states that market participants are not rational and are subject to multiple behavior biases, which further affect the decision-making. Some typical biases observed are overconfidence bias, herding bias

and status quo bias. Overconfidence bias occurs when market participants overestimate their intuitive ability and underestimate risk. Herding is when individuals follow the crowd's decision. Status quo bias refers to the tendency to stay in current status and unwillingness to make changes.

Financial advisors usually identify behavior biases from customer statements and question-askings in a conversation with them. The optimal suggestions will be given by taking the behavior biases into account. Besides, we believe more dynamic interests and subjective requirements can be observed in a conversation. Inspired by that, in this paper we propose a heterogeneous conversational recommender system (HConvoNet), which integrates unstructured conversation with structured profile and make more adaptive recommendations. In brief, our proposed framework consists of two modules: customer profile module and conversation module. The profile module aims to capture customer's important static needs, while the conversation module aims to extract behavior biases and dynamic interests. This is feasible since most companies have stored huge amount of conversation data from routine businesses, like telemarketing.

We model the structured profile data in a deep way, adopting DeepFM framework[5]. To capture the information embedded in a conversation comprehensively, we build the architecture using a two-level bidirectional Gated Recurrent Unit (GRU) with self-attention mechanism. The lower level encodes each single utterance and the upper level encodes the whole conversation considering contextual interactions among utterances.

We conduct the experiments on three internal datasets from Ping An Insurance, ESB, Wuyou and Anxin, which are popular insurance products in Ping An Insurance. In a conversation between insurance agent and customer, agent usually asks multiple questions in order to infer the insurance needs and preferences. The objective is to predict customer's purchase intention. The baseline models include industry popular methods and some variants of HConvoNet. Results show that our proposed model has a significant improvement over the baselines.

To summarize, we make the following main contributions:

- We propose an innovative heterogeneous conversational recommender system (HConvoNet) for financial products, which adapts customer behavior biases and dynamic interests.
- The proposed HConvoNet integrates structured customer profile data and unstructured conversation data and adopts cutting-edge NLP techniques.
- The proposed HConvoNet can be applied to most practical cases and has huge commercial value.

2 RELATED WORK

In this paper, we propose an innovative heterogeneous conversational recommender system (HConvoNet) for financial products. The most related domains are recommender system and textual information extraction. In this section, we will discuss the related work.

Recommender System. Factorization Machines[13] is a classical approach to model feature interactions using factorized parameters. Field-aware FM[9] is one of the variants of FM, which adds the field index into feature space. FNN[22], Wide & Deep[1] and DeepFM[5] are examples of using deep neural networks to learn more complex feature interactions. Deep learning techniques have also been applied to collaborative-filtering and content-based recommendation like [17] and [21]. [6] exploited RNN to develop a session-based recommender system. [19] uses RNN to build a recommender system for movie recommendation. Google develops a two-stage deep learning framework for YouTube video recommendation[3]. [18] and [23] propose hybrid models, which use deep learning to learn features of various domains.

However, most of the researches are exploiting the objective item/user nature and ignore unstructured data, which is subjective to user and affect the decision. There are some work focus on mining short text review to capture user sentiment like [12], but these approaches are not suitable for financial products.

Textual Information Extraction. Recurrent neural networks (RNN) is a standard way to extract sequential information. [14] extends RNN to a bidirectional RNN. [7] proposes the framework of long short-term memory (LSTM). Gated recurrent unit (GRU), proposed by [2] is seen as a better network to capture long sequential relationships. All these networks have succeeded in many natural language processing tasks. [11] uses LSTM for sentiment analysis. [24] proposes to use biLSTM to extract relationship. [4] uses biLSTM for speech recognition. [8] uses GRU for emotion recognitition and [20] uses GRU for document classification.

3 PROPOSED FRAMEWORK

3.1 Problem Definition

The dataset contains unstructured conversation transcripts and structured profile data, $D = \{C_s, P_s, y_s\}_{s=1}^{N}$, where N is the number of samples, C_s, P_s and y_s represent the conversation transcript, structured profile data and label of sample s respectively. Each conversation contains multiple utterances said either by the agent or customer, $C = \{u_i\}_{i=1}^{n}$, where u_i represents utterance i and n is number of utterances in the conversation. Each utterance consists of multiple words, $u_i = \{w_{i,j}\}_{j=1}^{K_i}$, where K_i is the number of words in utterance i. We aim to use heterogeneous data to predict customer's preference.

The overall architechture of our proposed framework can be seen in Figure 1. The framework can be explained in three parts: the profile module, conversation module and fusion part. We will clarify each one in the following content.

3.2 Profile Module

The profile module takes the form of DeepFM[5], which has two parts: FM part and DNN part.

FM part. FM is good at handling sparse data and can model the first-order impact and second-order interactions among all features. According to [13] and [5], FM can be expressed as:

$$y_{FM} = < \omega, x > + \sum_{j_1=1}^{d} \sum_{j_2=j_1+1}^{d} < V_i, V_j > x_{j1} \cdot x_{j2} \qquad (1)$$

where ω and V_i are parameters to estimate, $\omega \in \mathbb{R}^d$, $V_i \in \mathbb{R}^k$ (k is given as the feature embedding size) and $<,>$ is the dot product.

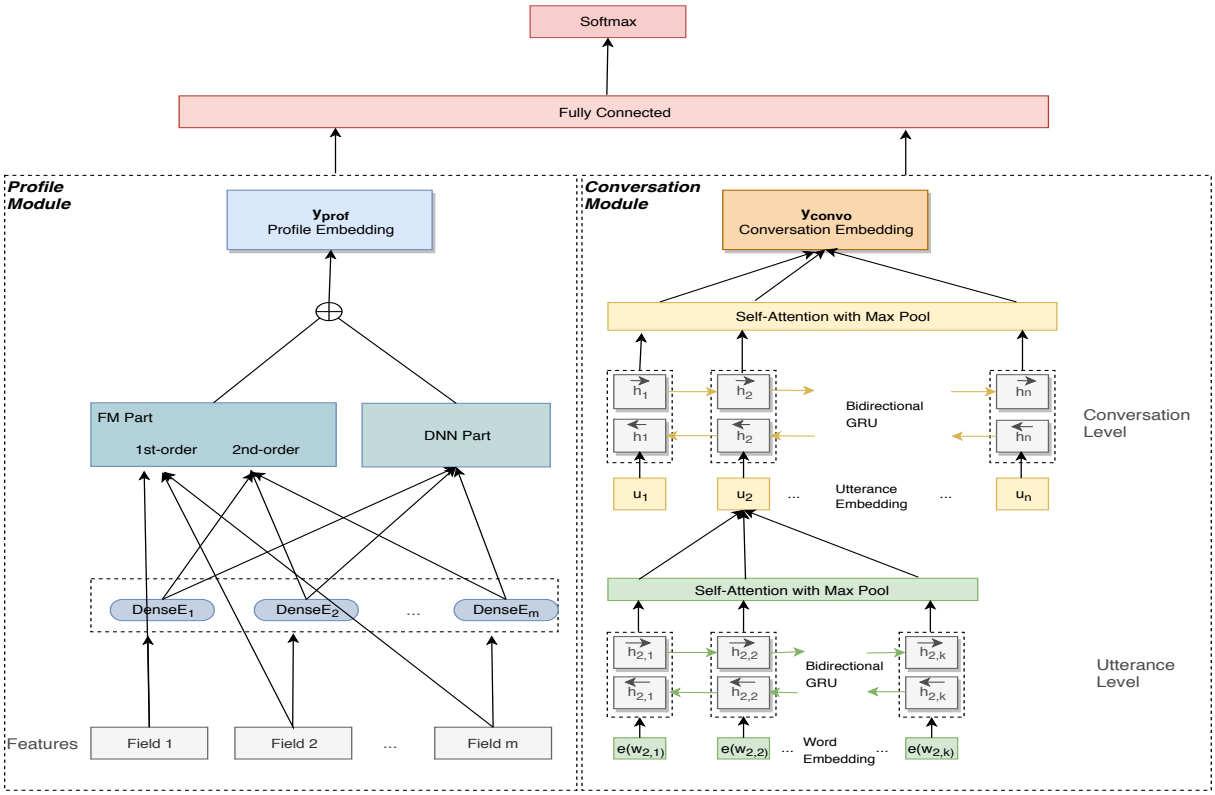

Figure 1: HConvoNet Framework

DNN part. DNN part models the more complex non-linear interactions among feature embeddings. Feed the output of embedding layer into the deep neural network and follow the forward process:

$$a^{l+1} = \sigma(W_f^l \cdot a^l + b_f^l) \tag{2}$$

where σ is the activation function, l is the layer depth, W_f is the weight matrix and b_f is the bias. We use Relu as the activation function and take output of the last layer a^L as DNN part representation y_{DNN}.

The final representation of the customer profile is the concatenation of both FM part and DNN part.

$$y_{prof} = [y_{FM}; y_{DNN}] \tag{3}$$

3.3 Conversation Module

The conversation module takes advantage of the cutting-edge natural language processing techniques. It can be seen as a two-level bidirectional GRU. The lower level encodes each single utterance and the upper level encodes the whole conversation considering contextual interactions among utterances. Besides, we propose the use of self-attention mechanism[16] to focus on more important information in utterance level embedding and conversation level embedding.

Utterance level. Suppose a single utterance u_i contains K_i words, $u_i = \{w_{i,j}\}_{j=1}^{K_i}$, where K_i is the number of words in utterance i. For each word $w_{i,j}$, we have:

$$\overrightarrow{h}_{i,j} = GRU(e(w_{i,j}), \overrightarrow{h}_{i,j-1}) \tag{4}$$

$$\overleftarrow{h}_{i,j} = GRU(e(w_{i,j}), \overleftarrow{h}_{i,j+1}) \tag{5}$$

where $e(w_{i,j})$ is the word embedding obtained from pre-trained word embeddings. The forward and backward hidden states are concatenated into $h_{i,j} = [\overrightarrow{h}_{i,j}; \overleftarrow{h}_{i,j}]$. Suppose the dimension of a unidirectional hidden state is m. Then $h_{i,j}$ has a dimension of $2m$.

We apply self-attention mechanism[16] to the concatenated hidden states to pay more attention to important words. We denote $H_i = (h_{i,1}; h_{i,2}; ...; h_{i,j}; ...; h_{i,K_i})$, where $H_i \in \mathbb{R}^{K_i \times 2m}$. The weight matrix in self-attention mechanism is calculated as :

$$A_i = softmax\left(\frac{H_i \cdot H_i^T}{\sqrt{2m}}\right) \tag{6}$$

where $A_i \in \mathbb{R}^{K_i \times K_i}$ and $\sqrt{2m}$ is a scale factor. The self-attended hidden states for words is then computed as:

$$H_i^{sa} = A_i \cdot H_i \tag{7}$$

where H_i^{sa} will have the same shape of H_i, which is $K_i \times 2m$. The single utterance embedding is then obtained by max-pooling over all words' self-attended hidden states:

$$e(u_i) = maxpool(H_i^{sa}) \tag{8}$$

where $e(u_i) \in \mathbb{R}^{2m}$.

Conversation level. We find the conversation embedding by a similar way as utterance embedding. Suppose a conversation consists of n utterances. We feed utterance embeddings obtained from the previous step into another bidirectional GRU:

$$\overrightarrow{h}_i = GRU(e(u_i), \overrightarrow{h}_{i-1}) \tag{9}$$

$$\overleftarrow{h}_i = GRU(e(u_i), \overleftarrow{h}_{i+1}) \tag{10}$$

We concatenate the forward and backward hidden states $h_i = [\overrightarrow{h}_i; \overleftarrow{h}_i]$ and represent all concatenated hidden states as a $n \times 2m$ matrix H.

Again, utterances are not of the same importance. We apply self-attention mechanism to learn the relative weights:

$$A = softmax\left(\frac{H \cdot H^T}{\sqrt{2m}}\right) \tag{11}$$

where $\sqrt{2m}$ is a scale factor. The self-attended hidden states matrix for utterances is then computed as:

$$H_i^{sa} = A \cdot H \tag{12}$$

The final conversation embedding is obtained by max-pooling over all utterances' hidden states:

$$y_{convo} = maxpool(H^{sa}) \tag{13}$$

3.4 Making Prediction

To generate the prediction, we concatenate the outputs from both profile module and the conversation module and feed into a Fully-Connected layer followed by a softmax function:

$$y_{final} = [y_{prof}; y_{convo}] \tag{14}$$

$$\hat{y} = softmax(W \cdot y_{final} + b) \tag{15}$$

The categorical cross-entropy is used as the loss function:

$$loss = -\sum_{i=1}^{N}\sum_{j=1}^{C} y_{i,j} log(\hat{y_{i,j}}) \tag{16}$$

where $y_{i,j}$ and $\hat{y_{i,j}}$ are the groundtruth and prediction.

4 EXPERIMENTS

4.1 Dataset

We conduct our experiments on three internal datasets from Ping An Insurance, ESB, Wuyou and Anxin. ESB, Wuyou and Anxin are three popular insurance products in Ping An Insurance. ESB is a kind of medical insurance. Wuyou is an universal insurance product, which has some investment feature. Anxin is an accident insurance. All three datasets contain unstructured conversation data and structured customer profile data. Labels are collected according to customer's purchase records after conversation within 15 days. The objective is to predict the customer's purchase intention, given his profile and conversation data. The time window of our datasets is May 2019. Due to the unbalanced distribution, we further downsample datasets to a rough ratio of 1:5. Table 1 provides the detailed information about each dataset. We randomly take 80% as the training set and 20% as the test set. We further partition the training set into development set and validation set with a 80/20 ratio.

Table 1: Summary of Datasets

Dataset	Pos[a]	Neg[b]	Feat[c]	avg. Utter[d]	avg. Convo Len[e]
ESB	2,349	12,258	15	12.23	139s
Wuyou	2,793	13,961	15	12.08	150s
Anxin	2,877	14,328	15	11.89	127s

[a] number of positive samples
[b] number of negative samples
[c] number of structured features
[d] average number of utterances in a conversation
[e] average length of a conversation in seconds

4.2 Data Preprocessing

We follow the general feature engineering process to preprocess the structured data. We preprocess the conversation data by the following steps: (1) We first extract the textual transcripts of the conversation audio using Automatic Speech Recognition (ASR) technique and clean the data due to some noises introduced by the previous step; (2) We segment each utterance into tokens by jieba package and add some business terminologies; (3) We remove all non-alphanumerics, stop words and the words with frequency lower than two; (4) We use the publicly available 300-dimensional *word2vec*[1] vectors trained on a large corpus across various domains. Words not in *word2vec* are randomly initialized.

4.3 Training Details

We use Pytorch to implement the experiments. We randomly shuffle the training set at the beginning of each epoch.

Parameters. For the profile module, we use 15 features (8 categorical and 7 numerical features) and set feature embedding size to 8. The DNN part has three layers and each layer has 300 neurons. For the conversation module, we set dimension of all hidden states to 300 and the utterance length to 40, which is around the average length of utterances.

Training. We adopt Adam[10] as the optimizer and set the initial learning rate to $2 * 10^{-4}$. An annealing strategy is utilized by decaying the learning rate by half every 20 epochs. For regularization purpose, we apply dropout with a rate of 0.5. Early stopping with a patience of 10 is adopted to terminate training based on the F-measure of the validation set.

4.4 Baselines

Our proposed model is compared with the following baselines, including traditional models and variants of HConvoNet:

- **Random Forest:** an ensembled tree-based algorithm.
- **DeepFM[5]:** DeepFM, use only structured profile data.
- **HConvoNet-GRU:** variant of HConvoNet, substitute the bidirectional GRU with unidirectional GRU.
- **HConvoNet-biLSTM:** variant of HConvoNet, substitute the bidirectional GRU with bidirectional LSTM.
- **HConvoNet-nsa:** variant of HConvoNet, no self-attention.
- **HConvoNet-meanpool:** variant of HConvoNet, substitute maxpool with meanpool.

[1] https://code.google.com/archive/p/word2vec/

4.5 Evaluation Metrics

We adopt F-measure as our evaluation metric. F-measure is the harmonic average of precision and recall and is often used for measuring performance in industry and many research fields.

4.6 Results

Table 2 shows the experimental results. We can see that HConvoNet outperforms the compared models on three datasets consistently, demonstrating the power of our proposed framework. Comparing variants of HConvoNet with traditional Random Forest and DeepFM, we notice variants of HConvoNet outperform Random Forest and DeepFM, reaffirming the significance of adding conversation data. Looking at the performances of variants of HConvoNet, we find HConvoNet-GRU is on a par with HConvoNet-biLSTM. HConvoNet-GRU has better performance on Anxin and poorer performance on ESB and Wuyou.

Table 2: Experimental Results

Model	ESB	Wuyou	Anxin
Random Forest	54.37	56.47	57.75
DeepFM	59.17	60.66	60.92
HConvoNet-GRU	64.79	66.71	66.53
HConvoNet-biLSTM	65.49	67.63	65.90
HConvoNet	**66.35**	**68.58**	**67.62**

Note: F-measure is used as the metric and is shown as a percentage.

We also test the impact of self-attention and different pooling method. Table 3 presents the performances on three datasets. We see HConvoNet achieves better performance over HConvoNet-nsa, indicating the effects of self-attention mechanism. Comparing meanpool and maxpool, we find that the difference is negligible. Our proposed HConvoNet succeeds in most cases.

Table 3: Variants of HConvoNet

Model	ESB	Wuyou	Anxin
HConvoNet-nsa	65.19	66.05	65.69
HConvoNet-meanpool	**66.38**	68.52	67.54
HConvoNet	66.35	**68.58**	**67.62**

Note: F-measure is used as the metric and is shown as a percentage.

5 CONCLUSIONS

In this paper, we propose an innovative heterogeneous conversational recommender system (HConvoNet) for financial products. We improve the traditional recommendation by integrating unstructured conversation data with structured profile data, thus considering customer static needs, behavior biases and dynamic interests. Future work could include exploring different methods to fuse heterogeneous data and involving multiple modalities of the conversation, like audio.

REFERENCES

[1] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Gregory S. Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & Deep Learning for Recommender Systems. In *DLRS@RecSys*.

[2] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *ArXiv* abs/1409.1259 (2014).

[3] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep Neural Networks for YouTube Recommendations. In *RecSys*.

[4] Alex Graves, Navdeep Jaitly, and Abdel rahman Mohamed. 2013. Hybrid speech recognition with Deep Bidirectional LSTM. *2013 IEEE Workshop on Automatic Speech Recognition and Understanding* (2013), 273–278.

[5] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. *ArXiv* abs/1703.04247 (2017).

[6] Balázs Hidasi, Massimo Quadrana, Alexandros Karatzoglou, and Domonkos Tikk. 2016. Parallel Recurrent Neural Network Architectures for Feature-rich Session-based Recommendations. In *RecSys*.

[7] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9 (1997), 1735–1780.

[8] Wenxiang Jiao, Haiqin Yang, Irwin King, and Michael R. Lyu. 2019. HiGRU: Hierarchical Gated Recurrent Units for Utterance-Level Emotion Recognition. In *NAACL-HLT*.

[9] Yu-Chin Juan, Yong Zhuang, Wei-Sheng Chin, and Chih-Jen Lin. 2016. Field-aware Factorization Machines for CTR Prediction. In *RecSys*.

[10] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014).

[11] Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Navonil Majumder, Amir Zadeh, and Louis-Philippe Morency. 2017. Context-Dependent Sentiment Analysis in User-Generated Videos. In *ACL*.

[12] G. Preethi, P. Venkata Krishna, Mohammad S. Obaidat, Vankadara Saritha, and Sumanth Yenduri. 2017. Application of Deep Learning to Sentiment Analysis for recommender system on cloud. *2017 International Conference on Computer, Information and Telecommunication Systems (CITS)* (2017), 93–97.

[13] Steffen Rendle. 2010. Factorization Machines. *2010 IEEE International Conference on Data Mining* (2010), 995–1000.

[14] Mike Schuster and Kuldip K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Trans. Signal Processing* 45 (1997), 2673–2681.

[15] H. Shefrin and Oxford University Press. 2002. *Beyond Greed and Fear: Understanding Behavioral Finance and the Psychology of Investing*. Oxford University Press. https://books.google.com/books?id=hX18tBx3VPsC

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *NIPS*.

[17] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2014. Collaborative Deep Learning for Recommender Systems. In *KDD*.

[18] Xinxi Wang and Ye Wang. 2014. Improving Content-based and Hybrid Music Recommendation using Deep Learning. In *ACM Multimedia*.

[19] Chao-Yuan Wu, Amr Ahmed, Alex Beutel, Alexander J. Smola, and How Jing. 2017. Recurrent Recommender Networks. In *WSDM*.

[20] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J. Smola, and Eduard H. Hovy. 2016. Hierarchical Attention Networks for Document Classification. In *HLT-NAACL*.

[21] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative Knowledge Base Embedding for Recommender Systems. In *KDD*.

[22] Weinan Zhang, Tianming Du, and Jun Wang. 2016. Deep Learning over Multi-field Categorical Data: A Case Study on User Response Prediction. *ArXiv* abs/1601.02376 (2016).

[23] Lei Zheng, Vahid Noroozi, and Philip S. Yu. 2017. Joint Deep Modeling of Users and Items Using Reviews for Recommendation. In *WSDM*.

[24] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification. In *ACL*.