# Beyond Polarity: Interpretable Financial Sentiment Analysis with Hierarchical Query-driven Attention

**Ling Luo**[1,4], **Xiang Ao**[1,4], **Feiyang Pan**[1,4], **Jin Wang**[2], **Tong Zhao**[3], **Ningzi Yu**[3] and **Qing He**[1,4]
[1]Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2]Computer Science Department, UCLA
[3]Deloitte China
[4]University of Chinese Academy of Sciences
{luoling18s, aoxiang, panfeiyang, heqing}@ict.ac.cn
jinwang@cs.ucla.edu, {tonzhao, jerryu}@deloitte.com.cn

## Abstract

Sentiment analysis has played a significant role in financial applications in recent years. The informational and emotive aspects of news texts may affect the prices, volatilities, volume of trades, and even potential risks of financial subjects. Previous studies in this field mainly focused on identifying polarity (e.g. positive or negative). However, as financial decisions broadly require justifications, only plausible polarity cannot provide enough evidence during the decision making processes of humanity. Hence an explainable solution is in urgent demand. In this paper, we present an interpretable neural net framework for financial sentiment analysis. First, we design a hierarchical model to learn the representation of a document from multiple granularities. In addition, we propose a query-driven attention mechanism to satisfy the unique characteristics of financial documents. With the domain specified questions provided by the financial analysts, we can discover different spotlights for queries from different aspects. We conduct extensive experiments on a real-world dataset. The results demonstrate that our framework can learn better representation of the document and unearth meaningful clues on replying different users' preferences. It also outperforms the state-of-the-art methods on sentiment prediction of financial documents.

## 1 Introduction

Sentiment analysis has been widely applied in financial applications since Robert Engle [Engle and Ng, 1993] suggested the asymmetric and affective impact of news on volatility. In recent years, researchers exploited various text resources e.g., news, microblogs, reviews, disclosures of companies to analyze the effects on markets in multifarious manners: impacting on price trends [Kazemian *et al.*, 2016], volume of trade [Engelberg and Parsons, 2011], volatilities [Rekabsaz *et al.*, 2017] and even potential risks [Nopp and Hanbury, 2015].

As a consequence, sentiment analysis in financial economics is considered different to that in user-product scenario [Liu, 2012; Ma *et al.*, 2017] of opinion mining. In particular, financial sentiment analysis aims to identify market confidence indicators of text from proxies, which are derived from human judgments or movements of the market. For example, a financial retrospective that contains positive sentiment is optimistic about the future financial prospects of a company. A news article is regarded to have negative sentiment if it reveals scandal of executives of a company since it increases the risks in the future of that company. The "Enron scandal" could be a more specific example[1]. Compared with users' reviews on products, finance related documents have these characteristics: verbose, redundant, and written in a style that makes them complex to process. Thus, it calls for specified framework to capture key information from financial documents.

Moreover, it is a challenging task to analyze sentiment in financial documents since the implicit sentiment in financial domain is derived from miscellaneous perspectives such as macroeconomic information, microstructure factors, event-oriented, company-specific. The sentiment derived from different perspectives could be rather different even if the representations are under similar lexicons and language structures. For example, in Figure 1 the word "rise" indicates a positive sentiment in the movement of prices which belongs to microstructure factors in Case 1. However, in Case 2, the same word contributes much less than the previous case as the phrase "non-performing loan ratio" dominates the negative sentiment from the event-oriented viewpoint.

Traditional approaches for financial sentiment analysis fail to model the complex structure of representations of financial texts very well because they are mainly based on domain-specific lexicons [Tsai and Wang, 2014; Nopp and Hanbury,

[1]https://en.wikipedia.org/wiki/Enron_scandal

| | |
|---|---|
| Case 1. Since the end of Aug, the stock price has **risen** about 30% | **Positive** |
| Case 2. The **non-performing loan ratio** has risen sharply to 2.35% | **Negative** |

Figure 1: An example of sentences in financial articles.

2015] or complicated feature engineering [Devitt and Ahmad, 2007; Schumaker and Chen, 2009; Rekabsaz *et al.*, 2017]. Recently, the deep neural network based methods [Ding *et al.*, 2015; Akhtar *et al.*, 2017] have also been applied in this field so as to learn the representations of documents. However, they still suffer from the defects in the following aspects:

Firstly, most of existing approaches only provide plausible polarity identifications (e.g. positive, negative or neutral). But in the practical applications, when people make financial decisions, the result of sentiment analysis is only one of the building blocks. In order to obtain a comprehensive analysis, explainable justifications rather than polarity labels are required from the model. Although deep learning based methods can do a good job in polarity prediction, they produce few clues of explanation.

Secondly, the sentiment of a same financial document can be varied for tasks from different departments. For example, an aggregated news bulletins of listed companies may be negative for company A while positive for company B and neutral for the market. It makes the overall polarity identification from document-level inapplicable for real-world application. Meanwhile, different users may concern about diverse aspects even for the same task. For instance, an accounting department will pay more attention to financing risks, e.g. debts, cash flow, etc., while a legal department will focus on legal risks, e.g. litigation, illegality, etc. Even if they simultaneously perform risk predictions for quoted companies, they may expect different results from the same document. Hence it is necessary to take diversity and personality of users' preference into account for financial sentiment analysis.

To address above problems, in this paper we propose an interpretable framework **FISHQA** (**FI**nancial **S**entiment analysis network with **H**ierarchical **Q**uery-driven **A**ttention) for financial sentiment analysis. First, we devise a hierarchical network structure to model the documents from multiple granularities: we first build representations of sentences from individual words and then aggregate those into document representations. Second, we equip FISHQA with a novel query-driven attention mechanism to meet the requirements from different aspects on one document. This mechanism is based on the idea that different parts of a document have varied weights in deciding the sentiment. To improve the flexibility of our model, we assign such weights according to the provided queries. Here the queries are natural language specified by the financial analysts from various departments. In this way, they can specify analysis results by deploying diverse queries for different purposes or tasks. With such mechanism, With such mechanism, our model is able to discover spotlights beyond the polarity of a document according to different users' requirements. Thus, compared with previous studies, our proposed method is more flexible and explainable. We conduct experiments on a real-world dataset related to bond default risk. The results demonstrate that FISHQA significantly outperforms the compared methods in polarity identifications as well as produce meaningful evidences for the prediction results. It can also pick out better representative information of the whole documents.

The remainder of this paper is organized as follows. Section 2 introduces the related work. We present the proposed FISHQA in Section 3. Experimental results are discussed in Section 4, the paper is concluded in Section 5.

## 2 Related Work

**Evaluating sentiments in financial articles.** There are basically two categories of evaluating sentiments in financial related articles.

First, the sentiment indicator is from proxies of financial subjects that can be derived or computed from the market, e.g. prices, volume of trade, volatilities and so on. [Ding *et al.*, 2015] proposed a neural network based framework to predict the stock price by measuring sentiment of events from financial news. [Nguyen and Shirai, 2015] predicted stock price movement by simultaneously analyzing topics and sentiments of social media. [Kazemian *et al.*, 2016] made some in-depth analysis on how to evaluate specific tasks of financial sentiment analysis. Their conclusion is consistent with our motivation. The intrinsic sentiments of financial documents can be diverse for different tasks, and even for the same task, they might be various among distinct perspectives.

Our work belongs to the second category, in which the sentiment is indicated by human specifications [Devitt and Ahmad, 2007; Akhtar *et al.*, 2017; Tetlock, 2007]. Here a sentiment analyzer is typically regarded as a potentially useful component to support either financial decision making tasks or specialized tasks where the indicators are difficult to be quantified, such as risk exposure management, bond rating. Lexicon-based and feature engineering approaches dominate the existing researches, which derives the incapable of modeling the semantic representations of text. Recently, [Akhtar *et al.*, 2017] proposed an ensemble learning approach and achieved state-of-the-art performance on fine-grained financial sentiment analysis. However, they are not able to produce explainable clues for either prediction or users' preferences.

**Sentiment analysis based on neural network.** [Kim, 2014] was the first to explore CNN for sentence-level text categorization tasks including sentiment classification. [Johnson and Zhang, 2017] utilized very deep word-level CNNs to capture global representations of texts. Different from those word-level models, [Zhang *et al.*, 2015] proposed a character-level CNN that achieved competitive results. [Tang *et al.*, 2015] performed document-level sentiment classification by leveraging user and product information.

The attention mechanism was firstly utilized in machine translation [Bahdanau *et al.*, 2015]. It then was adapted to both document-level [Yang *et al.*, 2016] and aspect-level [Ma *et al.*, 2017; Wang *et al.*, 2016] sentiment classification on user-product comments. Compared with them, our framework adopts more flexible attention mechanisms which are interactive with users and adjustable for different tasks.

Recently, memory network [Sukhbaatar *et al.*, 2015] was utilized to facilitate aspect-level sentiment analysis of user-product comments [Chen *et al.*, 2017; Tang *et al.*, 2016]. However, their methods cannot be adopted to our problem because it is difficult to find such a high-level abstract of "aspects" in financial domain. Actually, how to define a broadly accepted aspect in financial sentiment analysis still remains an open problem.

## 3 The FISHQA Model

In this section, we introduce our proposed model. The overall architecture of FISHQA is shown in Figure 2. It adopts a hierarchical structure that involves attention mechanism from multiple granularities: both in word level and sentence level.

In the real-world applications, analysts usually care about different aspects of financial articles for different purposes. For instance, when we concern more about uncertainty in personnel, "suicide" is more important than "decrease" though "decrease" may be usually informative to financial sentiment analysis. Hence, we may expect the words like "executives change" and "layoff" to be highly-weighted. Motivated by such observation, we devise the query-driven attention mechanism to highlight such preference-driven words in a sentence (and preference-driven sentences in document) by assigning higher attention weights to them. Next, we introduce the details of the FISHQA model and it consists of three layers: embedding layer, representation layer and output layer.

### 3.1 Embedding Layer

We represent a document with $n$ sentences as $d = \{s_1, s_2, ..., s_n\} \in \mathbb{D}$, where $\mathbb{D}$ is the document set and $s_t$ is the embedded representation. In the document, the $t$-th sentence consists of $l$ words $\{w_{t,1}, w_{t,2}, ..., w_{t,l}\} \in W$, where $t \in [1, \ldots, n]$ and $W$ is the set of word vocabulary. Each word $w_{t,i}$ is embedded into a real-value word-vector $x_{t,i} \in \mathbb{R}^D$ from embedding matrix $M_a \in \mathbb{R}^{V \times D}$, where $D$ is the dimension of word vectors and $V$ is vocabulary size. We set two sets of queries for word level and sentence level. We denote the $i$-th query for word level as $\{w'_{i,1}, w'_{i,2}, ..., w'_{i,l}\} \in W$, where $i \in [1, \ldots, k]$, and we embed it into a vector $q_i \in \mathbb{R}^D$ via the same matrix $M_a$. We generate the embedding vector of $j$-th query $p_j$ for sentence level in the same way and $j \in [1, \ldots, m]$. Here $k$ and $m$ represent the query numbers for word and sentence level, respectively. We represent $\{q_1, q_2, ..., q_k\}$ and $\{p_1, p_2, ..., p_m\}$ as two sets of embedded queries.

### 3.2 Sequence Encoder

In our model, we adopt Gated Recurrent Unit (GRU) [Bahdanau *et al.*, 2015] as the basic building block for sequence encoder. At step $t$, given input $x_t$ and previous hidden state $h_{t-1}$, the current hidden state $h_t$ can be updated by,

$$r_t = \sigma(U_r x_t + W_r h_{t-1} + b_r), \quad (1)$$

$$z_t = \sigma(U_z x_t + W_z h_{t-1} + b_z), \quad (2)$$

$$\tilde{h}_t = \tanh(U_h x_t + W_h(r_t \odot h_{t-1}) + b_h), \quad (3)$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t, \quad (4)$$

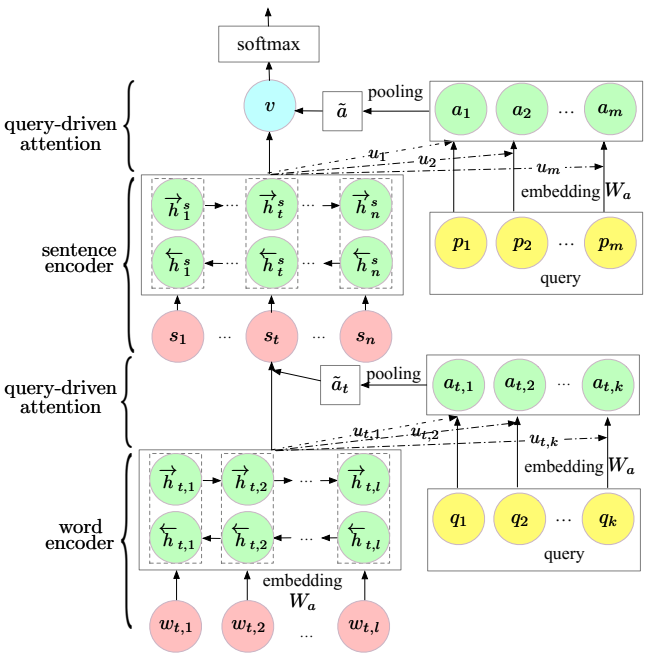


Figure 2: The network structure of FISHQA.

where $r_t$ represents the reset gate while $z_t$ is the update gate. $\sigma$ is the sigmoid activation function, $\odot$ stands for element-wise product. $U_r, U_z, U_h \in \mathbb{R}^{K \times D}$, $W_r, W_z, W_h \in \mathbb{R}^{K \times K}$, $b_r, b_z, b_h \in \mathbb{R}^{K \times 1}$ are parameters to be learned. Here $K$ is the dimension of hidden states, and $D$ is the input size. We argue that the sequence encoder in FISHQA is not limited by GRU, and other variations of RNN can also be utilized.

### 3.3 Hierarchical Query-driven Attention

Given a word $w_{t,i}$ from the $t$-th sentence, we embed it into a real-value vector $x_{t,i}$ with:

$$x_{t,i} = M_a w_{t,i}, \quad i = 1, \ldots, l.$$

Then we start from the word embedding and assign attention weights to enhance the representation from different levels.

**Word-level Query-driven Attention**

To better utilize the contextual information, we use bi-directional GRU to learn the hidden states, containing the forward representation $\overrightarrow{f}$ and backward $\overleftarrow{f}$ respectively,

$$h_{t,i} = \begin{bmatrix} \overrightarrow{h}_{t,i} \\ \overleftarrow{h}_{t,i} \end{bmatrix} = \begin{bmatrix} \overrightarrow{\text{GRU}}(x_{t,i}) \\ \overleftarrow{\text{GRU}}(x_{t,i}) \end{bmatrix}, \quad i = 1, \ldots, l, \quad (5)$$

where $h_{t,i}$ concatenates hidden states of the $i$-th word in the $t$-th sentence from both directions.

In our model, we adopt query-driven attention mechanism to select related words. Different queries are employed to consider various aspects of financial news. Following the previous work [Sukhbaatar *et al.*, 2015], for the $i$-th query $\{w'_{i,1}, w'_{i,2}, ..., w'_{i,l}\}$, we represent it with the vector $q_i$:

$$q_i = \sum_{j=1}^{l} M_a w'_{i,j}. \quad (6)$$

Then we construct the representation of $s_t$ as:

$$u_{t,i} = \tanh(W_{t,i}h_t + b_{t,i}), \quad i = 1, \ldots, k, \tag{7}$$

$$a_{t,i} = \text{softmax}({u_{t,i}}^{\mathrm{T}} q_i), \quad i = 1, \ldots, k, \tag{8}$$

$$\tilde{a}_t = \sum_{i=1}^{k} a_{t,i}/k, \quad s_t = \tilde{a}_t^{\mathrm{T}} h_t \tag{9}$$

where $h_t = (h_{t,1}, h_{t,2}, ..., h_{t,l})^{\mathrm{T}}$ denotes the combination of all the hidden states of the words in the $t$-th sentence, $u_{t,i}$ refers to the hidden representation of state $h_t$ through a fully-connected layer corresponding to query $q_i$ as shown in Eq. 7.

Each query is answered by selecting informative words in each sentence, that is, we aim to allocate high weights for related words. Given a query $q_i$, we measure the importance of words in sentence $s_t$ by an attention weight vector $a_{t,i}$, which can be computed as the inner product of query $q_i$ and $u_{t,i}$ followed by a softmax layer in Eq. 8. $\tilde{a}_t$ stands for the average of all the attention weight vectors (Eq. 9). We finally construct the $t$-th sentence $s_t$ with the sum of hidden states, weighted by $\tilde{a}_t$ in Eq. 9.

**Sentence-Level Query-driven Attention**

In sentence level, we learn the hidden states to form a document vector in the same way as the word-level. In particular,

$$h_t^s = \begin{bmatrix} \overrightarrow{h}_t^s \\ \overleftarrow{h}_t^s \end{bmatrix} = \begin{bmatrix} \overrightarrow{\text{GRU}}(s_t) \\ \overleftarrow{\text{GRU}}(s_t) \end{bmatrix}, \quad t = 1, \ldots, n,$$

where $h_t^s$ represents concatenated hidden states of GRU network from both directions in sentence $s_t$.

In sentence level, we also use query-driven attention mechanism to extract important sentences as well as form a better document representation with selected information.

$$u_j = \tanh(W_j h + b_j), \quad j = 1, \ldots, m, \tag{10}$$

$$a_j = \text{softmax}({u_j}^{\mathrm{T}} p_j), \quad j = 1, \ldots, m, \tag{11}$$

$$\tilde{a} = \sum_{j=1}^{m} a_j/m, \quad v = \tilde{a}^{\mathrm{T}} h, \tag{12}$$

where $h = (h_1^s, h_2^s, ..., h_n^s)^{\mathrm{T}}$ stands for the combination of all the hidden states of sentences, $p_j$ is the embedding vector of the $j$-th query raised for sentence level, $u_j$ is the hidden representation of state $h$ via a fully-connected layer for query $p_j$, and $a_j = (a_{j,1}, a_{j,2}, ..., a_{j,n})^{\mathrm{T}}$ represents the attention weight vector corresponding to $p_j$.

Similar to the word-level query-driven attention, each query is answered by selecting informative sentences in the document. Given a query $p_j$, we measure the importance of each sentence by the attention vector $a_j$, which is computed by the inner product of query $p_j$ and $u_j$ (Eq. 11), and allocate higher weights to related sentences. After averaging the attention weights for all the queries in sentence level (Eq.12), we can construct an overall representation of document $v$ by computing the weighted sum of hidden state $h$ (Eq. 12).

### 3.4 Output Layer

In the output layer, we feed the representative vector of the document into a fully-connected layer and predict the sentiment polarity. Here we use the softmax mechanism to get the probability distribution of the sentiment labels. The training objective is to minimize the cross-entropy loss as following:

$$L = -\sum_{d \in \mathbb{D}} \sum_{c=1}^{C} y_d^c \cdot \log(\hat{y}_d^c). \tag{13}$$

## 4 Experiments

### 4.1 Data Description

Our dataset combines a collection of $30,000$ documents, which were extracted from various Chinese mainstream financial websites over 30 days, ranging from May 26 to June 25, 2017. Among them, $7,648$ documents were annotated by three domain experts in the perspective that whether the corresponding bonds of the companies mentioned in the document will encounter the risk of default in the future. If the judgment is yes, the document is labeled as negative, otherwise as non-negative. Such manually labeled documents form our experimental set. The details are shown in Table 1.

### 4.2 Compared Methods

We compare our model with several baselines including feature-based methods and neural network based state-of-the-art approaches. Lexicon-based methods are not included in our experiments because few available sentiment lexicon suits our task and dataset. In addition, some previous work already demonstrated that deep learning-based methods outperforms [Yang *et al.*, 2016] lexicon-based methods.

1) **SVM+BoW** adopts SVM classifier using bag-of-word with frequency of each word as features.

2) **SVM+BoW TFIDF** adopts SVM classifier with TF-IDF of words as features under bag-of-word representation.

3) **CNN-word** is an available implementation[2] of the approach in [Kim, 2014].

4) **Bi-LSTM** learns document representation via a bidirectional LSTM network [Graves and Schmidhuber, 2005] by averaging all the hidden states of each word.

5) **LSTM-GRNN** was proposed by [Tang *et al.*, 2015] with hierarchical network structures.

6) **HAN** reported in [Yang *et al.*, 2016] adopts hierarchical networks with randomly initialized attention mechanism.

7) **FISHQA-$\mathbf{Q_1}$** represents FISHQA with a set of user-specific queries $Q_1$, which are raised by domain experts from the view of financial-related risk. We will detail the query construction later.

8) **FISHQA-$\mathbf{Q_2}$** is FISHQA with another user-specific query set $Q_2$ representing general focuses of financial documents.

| #docs | #non-neg docs | #neg docs | #vocabulary |
|---|---|---|---|
| 7,648 | 3,681 | 3,957 | 41,623 |
| #avg. sent. | #avg. words | #max sent. | #max words |
| 30.8 | 41.5 | 438 | 9,335 |

Table 1: Data Statistics.

| Query Set | $q_1$ | $q_2$ | $q_3$ |
|---|---|---|---|
| $\mathbf{Q_1}$ | **Financing** e.g.arrears | **Personnel** e.g.layoffs | **Litigation** e.g.lawsuit |
| $\mathbf{Q_2}$ | **Policy** e.g.announcement | **Company** e.g.industry sector | **Executives** e.g.CEO info |

Table 2: Query Set Information

[2]https://github.com/dennybritz/cnn-text-classification-tf

| Methods | Accuracy (%) | F1-score (%) | Acc-reduced (%) |
|---|---|---|---|
| SVM+BoW | 88.83 | 88.80 | - |
| SVM+BoW TFIDF | 89.57 | 89.64 | - |
| CNN-word | 88.48 | 89.52 | - |
| Bi-LSTM | 90.45 | 91.19 | - |
| LSTM-GRNN | 91.19 | 91.53 | - |
| HAN | 91.77 | 91.66 | 71.02 |
| FISHQA-$\mathbf{Q_1}$ | **94.46** | **94.49** | **86.65** |
| FISHQA-$\mathbf{Q_2}$ | 93.61 | 94.18 | 83.80 |

Table 3: Output Statistics: Acc-reduced denotes the classification accuracy of selected sentences.

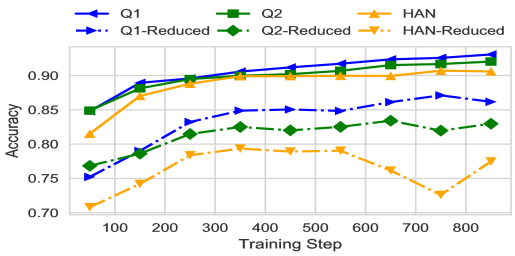


Figure 3: Comparison of FISHQA and HAN varying training steps.

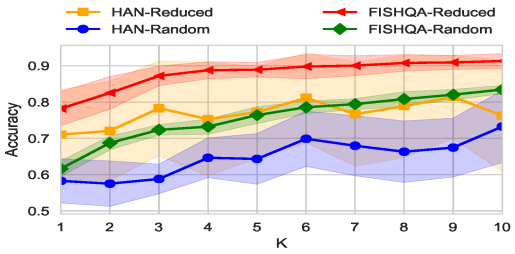


Figure 4: Comparison of FISHQA and HAN varying $K$.

## 4.3 Query Construction

The queries in FISHQA are manually created. Either sentences (e.g. "is there any non-performing loan?") or lists of concerned words (e.g. "arrears, debt, indebtedness") can be utilized as the queries.

Two groups of queries, namely $Q_1$ and $Q_2$, are compared in our experiments. As shown in Table 2, each query set contains three detailed queries and we denote them as $q_1$, $q_2$ and $q_3$ hereafter. The maximum and average query length are 45 and 20, respectively. The main difference between $Q_1$ and $Q_2$ is that $Q_1$ consists of customized risk-oriented queries from user-specific perspectives while $Q_2$ contains more general queries in financial domain. Due to the space limitation, we here show two small query sets as running examples. In real-world applications, however, we can construct question sets from different aspects upon users' needs and add any number of queries into each set. While we notice that our model also supports different query sets, in our experiments, we adopt the same query set for both word-level and sentence-level attention to magnify the effects of queries.

## 4.4 Experimental Settings

We perform the following preprocessing on the experimental dataset. First, we remove inserted advertising of each article. Then we use public NLP tools to split documents into sentences and tokenize each sentence into words with an extra financial exclusive lexicon. The lexicon mainly contains the names of listed-companies and the listed-bond names.

We set the dimension of word embedding as 200 and hidden size of GRU as 100. We optimize the training process using Adam [Kinga and Adam, 2015] with a mini-batch of size 64 and a learning rate 0.001. The number of words for each sentence and sentences for each document are set to be 45 and 30 by grid search, respectively. The word embeddings are trained with the models simultaneously, and all the other parameters of compared methods are fine-tuned to achieve the best results.

## 4.5 Experimental Results

We perform 5-fold cross-validation on the experimental dataset for all the methods. Table 3 reports the average accuracy and F1-score of the compared methods on the five folds. We observe that FISHQA-$Q_1$ reaches an accuracy of 94.46% and an F1-score of 94.49%, which outperforms HAN, the most competitive method in baselines, by 2.69% and 2.83%, respectively. The results of FISHQA-$Q_2$ also show similar trends. HAN performs better than other baselines since it picks out important information with the embedded hierarchical attentions. However, compared with our methods, HAN assigns the attention weights merely from the document itself. While FISHQA obtains the information of attention from queries provided by users, it can better decide the weights of words and sentences according to users' preferences. For instance, since the queries in $Q_1$ are related to potential risks of financial issues, FISHQA-$Q_1$ learns better document representations for predicting polarities of the task and thus achieves the best results.

Next, we evaluate the effectiveness of FISHQA on discovering informative sentences for a document $d$. We design a quantified measure as follows. For a document $d$, we choose the top $K$ representative sentences by the attention weights produced by the network, then we concatenate the extracted sentences as a new document $d'$ and feed it into the trained model to get sentiment prediction. Such derived prediction accuracy is denoted as Acc-reduced. Specifically, for FISHQA, we select the top $K$ sentences by concatenating the top $K_i$ weighted sentences associated to every query. It's worth mentioning that the number of selected sentences $K_i$ is allowed to be different when $K$ cannot be divisible by $m$. For HAN, we directly extract top $K$ highly-weighted sentences.

The fourth column of Table 3 demonstrates the average results on 5-fold cross-validation when $K$=2. We observe that FISHQA-$Q_1$ is much better at picking out more representative sentences with the domain-specific query-driven attentions. In more detail, FISHQA-$Q_1$ reaches 86.65% in Acc-reduced, and outperforms HAN by 15.63%. Though FISHQA-$Q_2$ takes a more general query set, it still has clear advantages by achieving 83.80% in Acc-reduced and outperforming HAN by 12.78%.

We further demonstrate the performances of FISHQA-$Q_1$, FISHQA-$Q_2$ and HAN on test sets at every 100 training steps as Figure 3. It shows that all the results are stable at around 0.9 after 300 steps. FISHQA-$Q_1$ starts to beat the others from step 400 while FISHQA-$Q_2$ begins to perform better than HAN from step 500. For the measure of Acc-reduced, both FISHQAs outperform HAN at almost each demonstrated step and gradually enlarge the gap. Moreover, it presents obvious

| FISHQA-$Q_1$ $q_1$ | FISHQA-$Q_1$ $q_2$ | FISHQA-$Q_1$ $q_3$ | Sentences | HAN $q$ |
|---|---|---|---|---|
| | | | **Bribery** incident triggered HEALTH Bioled temporarily suspended **stock**... | |
| | | | HEALTH Bioled announced to suspended the first **issue of shares**... | |
| | | | HEALTH Bioled **financial** worries, employees and subsidiaries trap in **bribery** scandal | |
| | | | CSFS will carefully **investigate** the involved matters questioned by media... | |
| | | | Huang of HEALTH Bioled was **suspected of bribery** case and other matters... | |
| | | | HEALTH Bioled was **suspected to give rebate** to Fourth People's Hospital... | |

Table 4: Case study between FISHQA-$Q_1$ and HAN. The details of queries' categories in $Q_1$ are referred to Table 2.

| FISHQA-$Q_1$ $q_1$ | FISHQA-$Q_1$ $q_2$ | FISHQA-$Q_1$ $q_3$ | Sentences | FISHQA-$Q_2$ $q_1$ | FISHQA-$Q_2$ $q_2$ | FISHQA-$Q_2$ $q_3$ |
|---|---|---|---|---|---|---|
| | | | Liren Group's financial officers admited a total of 2.2 billion yuan of **private loans**... | | | |
| | | | Wenzhou Liren Education Group announced no more acceptance of **private borrowing**... | | | |
| | | | the CEO had twice tried to **commit suicide** because of **debt** more than 20 billion | | | |
| | | | he confirmed the chairman Dong Shun-sheng has committed **suicide attempt** twice... | | | |
| | | | Political, legal and financial joint **investigation**, maintenance of teaching order | | | |
| | | | Taishun County has set up an **investigation team** to conduct a thorough investigation | | | |

Table 5: Case study between FISHQA-$Q_1$ and FISHQA-$Q_2$. The details of queries' categories in $Q_1$ and $Q_2$ are referred to Table 2.

and stable upward tendency for FISHQAs while HAN takes on fluctuation. These observations confirm that the proposed model can stably select better informative sentences to represent original documents.

We conduct another experiment to test the performances on Acc-reduced by varying the number of selected sentences $K$. Additionally, we compare the contributions of highly weighted sentences with randomly selected ones for both FISHQA and HAN. In Figure 4, we name the average accuracy on Acc-reduced of FISHQA-$Q_1$ as FISHQA-Reduced and that of randomly selected sentences as FISHQA-Random. The same notation also works for HAN. For each method, we conduct 50 times on each $K$, where $K$=$[1, \ldots, 10]$, and we visualize the average trends as well as the standard deviation, which reflects the distributions of Acc-reduced on the 50 rounds, in Figure 4. The results indicate that:

First, with the growth of the number of selected sentences, Acc-reduced of FISHQA takes on a stable trend of rising and a small variance. Additionally, the results are comparable with the accuracy on the full documents when selecting more than 5 sentences. Yet Acc-reduced of HAN shows fluctuated rising and larger variance. Because the uncertainty of randomly initialized vector used by HAN determines that it cannot always find the representative information.

Second, the selected highly-weighted sentences of both FISHQA and HAN perform better than randomly chosen ones. Meanwhile, FISHQA-Random takes on a trend of rising with a smaller variance and higher accuracy than HAN-Random at every $K$. We conjecture the reason is that query-driven attention mechanism can deal with broader information as various queries are able to learn multiple aspects of a document. Hence, FISHQA can obtain more effective information even from randomly selected sentences.

### 4.6 Case Study on Interpretability

Finally, we compare the results of the attention mechanisms of FISHQA-$Q_1$, FISHQA-$Q_2$ and HAN by visualizing the weights of sentences and words in the same documents as shown in Table 4 and 5. For each table, the depth of color for each column aside illustrates the sentence-level distributions of attention weights related to each query. The colored words represent highly-weighted ones through word-level query-driven attention mechanism for the model in the left column, while the underlined words are for the model in the right column. For both word and sentence, the darker the color is, the higher attention weight it has.

First, we compare the differences of attention mechanisms between FISHQA-$Q_1$ and HAN in Table 4. For example, the first and third sentence have high attention weights for query $q_2$ of FISHQA-$Q_1$. They all tell information about "bribery cases", which are clearly related to personnel perspective. While for HAN, its attention weights on sentences are evenly distributed. HAN tends to focus on sentences containing emotive words, such as "worries" and "suspended". Yet the financial sentiments are task-specific and more than only subjective emotive languages. On the other hand, unlike our method, HAN cannot provide any explainable clues according to user's preference.

Second, we visualize the attention results of FISHQA with query set $Q_1$ and $Q_2$ as Table 5. There are huge differences among the sentences and words that are paid attention to. Recall that the queries in $Q_1$ are customized risk-oriented ones, while those in $Q_2$ are general concepts in financial domain. The FISHQA equipped with $Q_1$ pays more attention to sentences containing words like "debt" (replying to financing, i.e. $q_1 \in Q_1$), "suicide attempt" (replying to personnel, i.e. $q_2 \in Q_1$), and "investigation" (replying to litigation, i.e. $q_3 \in Q_1$). However, the model with $Q_2$ pays more attention to company names, e.g. "Liren Group", and title of executives, e.g. "CEO" and "chairman", which are informative indicators to the viewpoints that user concerns. Such results demonstrate that our FISHQA model can effectively find the important parts within a document according to users' queries.

## 5 Conclusion

In this paper, we proposed an attention driven model FISHQA to conduct document-level financial sentiment analysis. We design a hierarchical structure that learns the representation of document from multiple granularities. We also propose a query-driven attention mechanism in different levels to discover spotlights from texts according to user-specified queries and provide explainable results. Experimental results on a real-world financial dataset demonstrate the effectiveness of FISHQA. It also suggests significant superiority in selecting informative and explainable information.

## Acknowledgements

This research is supported by National key R&D program of China (No.2017YFB1002104), the National Natural Science Foundation of China (No.61602438, 91546122, 61573335), and Guangdong provincial science and technology plan projects (No. 2015B010109005).

## References

[Akhtar *et al.*, 2017] Md Shad Akhtar, Abhishek Kumar, Deepanway Ghosal, Asif Ekbal, and Pushpak Bhattacharyya. A multilayer perceptron based ensemble technique for fine-grained financial sentiment analysis. In *EMNLP*, pages 540–546, 2017.

[Bahdanau *et al.*, 2015] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015.

[Chen *et al.*, 2017] Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. Recurrent attention network on memory for aspect sentiment analysis. In *EMNLP*, pages 463–472, 2017.

[Devitt and Ahmad, 2007] Ann Devitt and Khurshid Ahmad. Sentiment polarity identification in financial news: A cohesion-based approach. In *ACL*, pages 984–991, 2007.

[Ding *et al.*, 2015] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Deep learning for event-driven stock prediction. In *IJCAI*, pages 2327–2333, 2015.

[Engelberg and Parsons, 2011] Joseph E Engelberg and Christopher A Parsons. The causal impact of media in financial markets. *The Journal of Finance*, 66(1):67–97, 2011.

[Engle and Ng, 1993] Robert F Engle and Victor K Ng. Measuring and testing the impact of news on volatility. *The Journal of Finance*, 48(5):1749–1778, 1993.

[Graves and Schmidhuber, 2005] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5-6):602–610, 2005.

[Johnson and Zhang, 2017] Rie Johnson and Tong Zhang. Deep pyramid convolutional neural networks for text categorization. In *ACL*, pages 562–570, 2017.

[Kazemian *et al.*, 2016] Siavash Kazemian, Shunan Zhao, and Gerald Penn. Evaluating sentiment analysis in the context of securities trading. In *ACL*, pages 2094–2103, 2016.

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. In *EMNLP*, pages 1746–1751, 2014.

[Kinga and Adam, 2015] D Kinga and J Ba Adam. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[Liu, 2012] Bing Liu. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167, 2012.

[Ma *et al.*, 2017] Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. Interactive attention networks for aspect-level sentiment classification. In *IJCAI*, pages 4068–4074, 2017.

[Nguyen and Shirai, 2015] Thien Hai Nguyen and Kiyoaki Shirai. Topic modeling based sentiment analysis on social media for stock market prediction. In *ACL*, pages 1354–1364, 2015.

[Nopp and Hanbury, 2015] Clemens Nopp and Allan Hanbury. Detecting risks in the banking system by sentiment analysis. In *EMNLP*, pages 591–600, 2015.

[Rekabsaz *et al.*, 2017] Navid Rekabsaz, Mihai Lupu, Artem Baklanov, Allan Hanbury, Alexander Dür, and Linda Anderson. Volatility prediction using financial disclosures sentiments with word embedding-based ir models. In *ACL*, pages 1712–1721, 2017.

[Schumaker and Chen, 2009] Robert P Schumaker and Hsinchun Chen. Textual analysis of stock market prediction using breaking financial news: The azfin text system. *ACM TOIS*, 27(2):12, 2009.

[Sukhbaatar *et al.*, 2015] Sainbayar Sukhbaatar, arthur szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In *NIPS*, pages 2440–2448, 2015.

[Tang *et al.*, 2015] Duyu Tang, Bing Qin, and Ting Liu. Document modeling with gated recurrent neural network for sentiment classification. In *EMNLP*, pages 1422–1432, 2015.

[Tang *et al.*, 2016] Duyu Tang, Bing Qin, and Ting Liu. Aspect level sentiment classification with deep memory network. In *EMNLP*, pages 214–224, 2016.

[Tetlock, 2007] Paul C Tetlock. Giving content to investor sentiment: The role of media in the stock market. *The Journal of Finance*, 62(3):1139–1168, 2007.

[Tsai and Wang, 2014] Ming-Feng Tsai and Chuan-Ju Wang. Financial keyword expansion via continuous word vector representations. In *EMNLP*, pages 1453–1458, 2014.

[Wang *et al.*, 2016] Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. Attention-based lstm for aspect-level sentiment classification. In *EMNLP*, pages 606–615, 2016.

[Yang *et al.*, 2016] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J. Smola, and Eduard H. Hovy. Hierarchical attention networks for document classification. In *HLT-NAACL*, pages 1480–1489, 2016.

[Zhang *et al.*, 2015] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *NIPS*, pages 649–657, 2015.