future internet

MDPI

Article

Forward-Looking Element Recognition Based on the LSTM-CRF Model with the Integrity Algorithm

Dong Xu, Ruping Ge * and Zhihua Niu

School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China; dxu@shu.edu.cn (D.X.); ZhNiu@shu.edu.cn (Z.N.)

* Correspondence: geruping@shu.edu.cn; Tel.: +86-188-1760-8630

Received: 31 October 2018; Accepted: 8 January 2019; Published: 14 January 2019

Abstract: A state-of-the-art entity recognition system relies on deep learning under data-driven conditions. In this paper, we combine deep learning with linguistic features and propose the long short-term memory-conditional random field model (LSTM-CRF model) with the integrity algorithm. This approach is primarily based on the use of part-of-speech (POS) syntactic rules to correct the boundaries of LSTM-CRF model annotations and improve its performance by raising the integrity of the elements. The method incorporates the advantages of the data-driven method and dependency syntax, and improves the precision rate of the elements without losing recall rate. Experiments show that the integrity algorithm is not only easy to combine with the other neural network model, but the overall effect is better than several advanced methods. In addition, we conducted cross-domain experiments based on a multi-industry corpus in the financial field. The results indicate that the method can be applied to other industries.

Keywords: LSTM-CRF model; elements recognition; linguistic features; POS syntactic rules

1. Introduction

In recent years, thanks to the broad application of natural language processing (NLP) technology, financial information processing capabilities have been unprecedentedly improved. For example, studies have explored the effects of financial events on stock price predictions [1,2], used firm reports to predict corporate performance [3], performed financial text sentiment analysis [4], and identified and extracted financial concepts (financial named entities, FNE) [5]. Notably, firm research reports have always been an important source of financial information. This information can be used to analyze the recent situation of a company from a professional perspective, make predictive assessment regarding the economic patterns and trends of the company in upcoming years, and provide professional investment advice. Previous studies [6] have also noted that compared to individual subjective texts, such as stocks and forums, the firm research reports are more realistic, and can provide a reliable source for financial text sentiment calculations and financial early warning decisions.

In 1980, the American Stock Exchange required all listed companies to include a Management Discussion and Analysis (MD&A) section in their annual reports. MD&A focuses on the disclosure of forward-looking statements (FLSs) that may have a significant impact on the company. An FLS is an assessment and expectation of the future development trends and prospects of the company, and such information is of great significance to venture investors and other stakeholders. In recent years, many scholars have analyzed and thoroughly explored FLSs. For example, Feng Li [7] used a naive Bayes classifier to explore the correlation between the information contained in FLSs in annual reports and economic factors. In article, our objective is to achieve the fine-scale recognition of forward-looking sentences in research reports; for example, "原料药业务贡献的净利润将显著增厚" (The net profit contributed by the drug substance business will increase significantly). The element

triple <Entity, Attribute, Attribute value> corresponds to <原料药业务 (drug substance business), 净利润 (net profit), 显著增厚 (significant increase)> in the sentence. This basic task can be applied in high-level applications such as sentiment analysis, investment decision making, and stock forecasting.

Difficulties encountered in our work mainly include the following points. First, unlike other fields elements, finance is a relatively open field. Entities or attributes are very broad and difficult to identify. Second, new entities are constantly emerging. It is difficult to develop simple template rules that apply to all situations. Third, in the financial corpus, attributes, and attribute value elements appear mainly in the form of compound words or clauses. Therefore, the integrity of the recognition of the elements must also be considered. Deep learning can be used to overcome the two problems of open fields and new entities. In particular, deep learning is a data-driven approach that automatically identifies elements and builds models based on learning appropriate to the field. However, natural language itself is a highly symbolic and complex discrete rule, it is a collection of conventions and domain knowledge associated with the process of human evolution. In a purely data-driven approach, the language may lose its original meaning. Therefore, we combine deep learning with linguistic features and propose the LSTM-CRF model with the integrity algorithm, mainly to improve the recognition effect by correcting the boundaries of LSTM-CRF model annotations. This method combines the advantages of data-driven methods and dependency grammar to improve the accuracy and recall of elements.

The structure of this paper is as follows. Section 2 introduces the status of element recognition research. Section 3 describes forward-looking element recognition based on the LSTM-CRF model with the integrity algorithm. Section 4 details the experimental steps and the analysis of the experimental results. The Section 5 is the conclusion of the paper and outlooks for future work.

2. Related Work

The research objective of this article is similar to that of Feldman et al. [8] The goal is to obtain various elements of specified type at the sentence level, including entities, attributes, and attribute values. The existing element recognition methods can be largely divided into the following three categories.

2.1. Rule/Dictionary Methods

Feldman et al. [8] used regular expressions to extract product names and attributes in the construction of running shoes based on an automotive product brand dictionary and attribute dictionary. Another study [9] used association rules to mine frequent nouns or noun phrases in product reviews as features of recognition. Moreover, Reference [10] proposed a rule-based recognition strategy based on the syntactic relationships among opinion words and the target, and they expanded the initial opinion lexicon and extracted the target using a propagation algorithm.

2.2. Machine Learning

Hai et al. [11] extracted a maximum entropy model for ontological event elements to obtain the five tuples in comparative sentences, namely, the comparison subject, comparison object, comparison attribute, comparison word, and evaluation word. The CRF model does not rely on independence assumptions; therefore, it can be used to fuse complex non-local features and better capture potential relationships among states. Therefore, it is widely used in entity recognition tasks. A previous study [12] proposed the combination of domain knowledge and vocabulary information using the CRF model to identify product features. Finkel [13] proposed an automatic tagging model that considers the characteristics of words, including suffixes, part-of-speech sequences and the morphology of words. Choi [14] used the CRF model, fusion words, part-of-speech features, opinion lexicon features, and dependency tree features to assess recognition from a specific viewpoint.

2.3. Deep Learning Methods

With the development of word-distributed representation, deep learning is a powerful tool for sequence modeling. Typically, a word is expressed by word embedding and then input into the neural network model. In this process, the hidden layer of the multilayer structure is used to extract features and predict the label. Collobert et al. [15] combined a convolutional neural network (CNN) with a CRF to achieve better results for named entity recognition tasks. Huang et al. [16] presented a bidirectional LSTM-CRF model (Bi-LSTM-CRF model) for NLP benchmark sequence tagging data. The model yielded an F-value of 88.83% for the CONLL2003 corpus. Limsopatham et al. [17] used the Bi-LSTM framework to automatically learn orthographic features and investigate the named entity recognition problem. The proposed approach performed excellently in "segmentation and categorization" and "segmentation only" subtasks. Lample et al. [18] constructed a LSTM-CRF model using word representations and character-based representations of captured morphological and orthogonal information, and the model performed well in named entity recognition (NER) tasks in four languages.

In the above studies, the rule methods are effective, but they require manual construction and domain knowledge. Machine learning models require large numbers of corpus-based and manual features, and the recall rate is not high. Although deep learning models provide satisfactory recognition effects, they ignore the meaning of the language itself and cannot be further improved. Analyses of financial domain corpora have shown that these methods cannot be directly applied to the financial field. Notably, the openness and integrity of elements must be considered.

In recent years, dependency grammar has performed well in identifying tasks. Popescu [19] and others used the rules of dependency syntax to recognize emotional words. Bloom et al. [20] used the rules of syntactic dependency to formulate rules and identify evaluation objects and evaluation word pairs. Somprasertsri et al. [21] used dependency syntax to build evaluation objects and candidate sets of evaluation words and obtain evaluation objects and evaluation words. Then, the candidate set was reselected using a maximum entropy model to obtain the final evaluation object and evaluation word pair. Therefore, we can rely on the syntax integrity algorithm to correct the element boundaries and combine it with the LSTM-CRF model to propose the LSTM-CRF model with the integrity algorithm. This method has the following three advantages over the conventional method:

1. The dependency syntax can capture the long-range collocation and modification relations between the internal components of a sentence. This feature can be used to solve the problem of integrity;
2. Open-element and new entity issues can be solved with the superior recognition performance of the LSTM-CRF model;
3. The proposed model meets the requirement of simultaneously identifying different types of elements.

3. LSTM-CRF Model with the Integrity Algorithm

The proposed element recognition method based on the LSTM-CRF model is mainly composed of the LSTM-CRF model and integrity algorithm. This section starts by giving an overview of our model. Next, we will describe these two parts of the proposed method in detail.

3.1. Model Overview

The model architecture is showed in Figure 1. It first uses the LSTM-CRF model to obtain the tag sequence (LS_1, LS_2 ... LS_n) under the influence of a data drive. The input is a word vector corresponding to a word, and the word vector can be pre-trained or trained together in the model. The output is the label for each word. This process can only get a rough range of each element. The most important idea of the LSTM-CRF model is to add the CRF model as the decoding layer of the model based on Bi-LSTM and consider the rationality between the prediction results. To further obtain

accurate boundaries, the integrity algorithm formed by the POS syntactic rules is used to continuously correct the range of element recognition to obtain the final tag sequence (L_1, L_2 ... L_n).

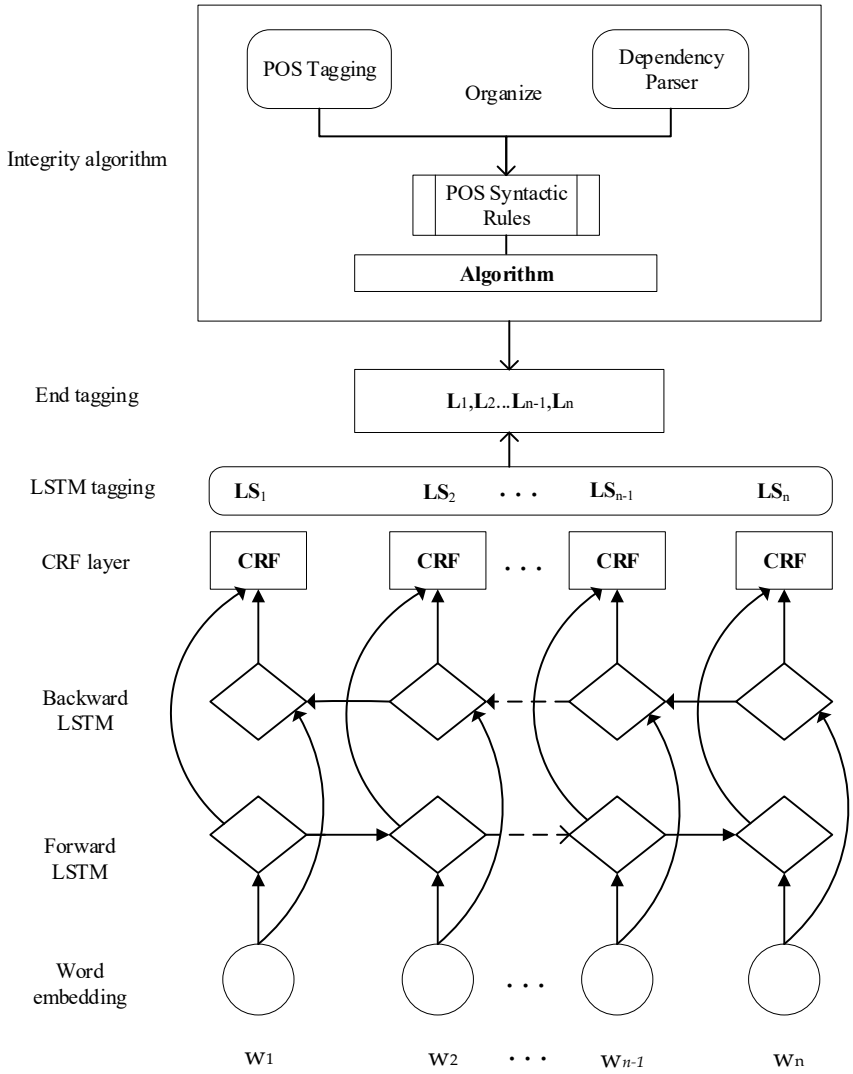

Figure 1. The main architecture of our model.

3.2. LSTM-CRF Model

LSTM networks are a specific form of recurrent neural network (RNN) proposed by Sepp Hochreiter, Jurgen Schmidhuber, and colleagues [22]. Unlike the general RNN model, conventional neurons are replaced with memory cells, each of which consists of input gates, forget gates, and output gates. This approach not only allows long-term dependencies to be identified but also avoids the problem of gradient disappearance or gradient expansion. In this article, the LSTM-CRF model is like that proposed by Huang et al. [16]. The model can be formulated as follows:

$$\boldsymbol{i}_t = \sigma\left(\boldsymbol{W}_{xi}\,\boldsymbol{x}_t + \boldsymbol{W}_{hi}\,\boldsymbol{h}_{t-1} + \boldsymbol{W}_{ci}\,\boldsymbol{c}_{t-1} + \boldsymbol{b}_i\right) \tag{1}$$

$$\boldsymbol{f}_t = \sigma\left(\boldsymbol{W}_{xf}\,\boldsymbol{x}_t + \boldsymbol{W}_{hf}\,\boldsymbol{h}_{t-1} + \boldsymbol{W}_{cf}\,\boldsymbol{c}_{t-1} + \boldsymbol{b}_f\right) \tag{2}$$

$$\tilde{c} = tanh\left(\boldsymbol{W}_{xc}\boldsymbol{x}_t + \boldsymbol{W}_{hc}\boldsymbol{h}_{t-1} + \boldsymbol{b}_c\right) \tag{3}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tilde{c} \tag{4}$$

$$\boldsymbol{o}_t = \sigma\left(\boldsymbol{W}_{xo}\,\boldsymbol{x}_t + \boldsymbol{W}_{ho}\,\boldsymbol{h}_{t-1} + \boldsymbol{W}_{co}\,\boldsymbol{c}_t + \boldsymbol{b}_o\right) \tag{5}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot tanh(\boldsymbol{c}_t) \tag{6}$$

where f, i, and o represent the forget, input, and output gates, respectively; c_{t-1} represents the state of the cell at time $t-1$; c_t represents the state of the cell at time t. h_t represents the output of the current state; h_{t-1} represents the output of the unit at the previous moment; σ is the logistic sigmoid function; W and b represent the weight and bias, respectively; and $\odot$ is the element-wise product.

The CRF model is an undirected graph model that was proposed by Lafferty et al. [23] in 2001. The model has obvious advantages in labeling and segmenting serialized data. It directly models the conditional distribution of data and can effectively avoid the problem of mark biasing experienced by other discriminant models. Moreover, the CRF model is different from other production models because it does not rely on the independence assumption, which allows it to fuse a variety of complex, non-local features and better capture the potential relationships among states.

The goal of the CRF model is to calculate the conditional probability distribution of the optimal output sequence (predictive sequence) given the input sequence (observed sequence), i.e., $P(Y \mid X)$. For example, assume that the random variable $X = (x_1, x_2, \ldots, x_{n-1}, x_n)$ is the input sequence and the random variable $Y = (y_1, y_2, \ldots, y_{n-1}, y_n)$ is the output sequence. Then, under the condition that the observation sequence is X, the conditional probability distribution of the prediction sequence Y is as shown in Formulas (7) and (8):

$$P(X|Y) = \frac{1}{Z(X)} \exp(\sum_k \lambda_k \alpha_k(y_{i-1}, y_i, x, i) + \sum_k \mu_k \beta_k(y_i, x, i)) \tag{7}$$

$$Z(X) = \sum_y \exp(\sum_k \lambda_k \alpha_k(y_{i-1}, y_i, x, i) + \sum_k \mu_k \beta_k(y_i, x, i)) \tag{8}$$

where α_k and β_k are binary functions (0 or 1); α_k is a transfer eigenfunction that reflects the correlation between adjacent marker variables at two positions, $i-1$ and i, and the effect of observation sequences on marker variables; β_k is a state eigenfunction that represents the effect of a node at observation sequence position i on the marker variable; λ_k and μ_k are parameters corresponding to the characteristic functions α_k and β_k, respectively; and $Z(X)$ is the normalization factor.

The LSTM-CRF model is widely used in the NER task. The powerful data fitting ability of the LSTM model can be utilized, and sequence labeling can be directly implemented by the CRF. The labeling of each current word is related to the labeling result at the previous moment. The LSTM-CRF model consists of a word embedding layer, a bidirectional LSTM layer, and a CRF layer. The first word embedding layer uses pretraining to map each word w_i of a sentence into a d-dimensional word vector. The LSTM-CRF model architecture is shown in Figure 2.

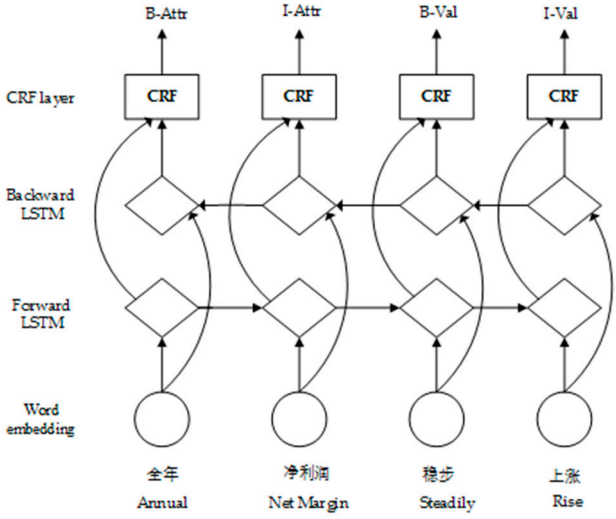

Figure 2. LSTM-CRF model architecture.

The word vector is input into the bidirectional LSTM layer, and the hidden state of the forward LSTM output and the hidden state of the backward LSTM output are obtained. The hidden state sequence output by each position is spliced by position to obtain a complete hidden state sequence $h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}]$.

Next, P is regarded as the matrix of scores outputted by the LSTM network of size $n \times k$, where k indicates the category of the tags. $P_{i,j}$ corresponds to the score of the j-th tag of the i-th word in a sentence. Given an input sentence $X = (x_1, x_2, \ldots, x_n)$ and a sequence of predictions, $y = (y_1, y_2, \ldots, y_n)$, the associated score is defined as follows:

$$score(X, y) = \sum_{i=1}^{n} P_{i,yi} + \sum_{i=1}^{n} A_{yi-1,yi} \qquad s.t.P \in R^{n \times k}, A \in R^{(k+2) \times (k+2)} \tag{9}$$

where $A_{i,j}$ is the transfer score from the ith tag to the jth tag. The start and end tags are added to the set of possible tags, and these tags, namely, y_0 and y_n, denote the start and end of a sentence. Thus, A is a square matrix of size $k + 2$. After applying a softmax function to all possible tag sequences, the probability of the sequence y is as follows:

$$P(y|X) = \frac{\exp(score(X, y))}{\sum_{y\prime} \exp(score(X, y\prime))} \tag{10}$$

The model is trained by maximizing the log likelihood function:

$$\log P(y^x|X) = score(X, y^x) - \log(\sum_{y\prime} score(X, y\prime)) \tag{11}$$

Finally, the Viterbi algorithm is used to solve the optimal path:

$$y^* = \arg\max_{y\prime} score(X, y\prime) \tag{12}$$

3.3. Integrity Algorithm

3.3.1. Dependency Syntax

To improve the integrity of element recognition, the dependency syntax relationship is introduced, which can capture the long-range collocation and modification relationships. This relationship is combined with the part-of-speech (POS) rules to filter out useless rules that occasionally occur. Then, the POS syntactic rules are organized to identify the complete structure of the elements. The LTP dependency syntax analysis based on the cloud platform (https://www.ltp-cloud.com/), each dependency is composed of core words and modifiers. The dependency relationship between two words is connected by a dependency arc, and the specific relationship between collocations is indicated by the marks on the dependency arc. Tags are shown in Table 1.

Table 1. Tags of LTP-cloud Platform Dependency Syntax Analysis.

Tags	Dependency Relations	Tags	Dependency Relations
SBV	Subject-verb	CMP	Complement
VOB	Verb object	COO	Coordinate
FOB	Fronting object	POB	Preposition object
DBL	Double	LAD	Left adjunct
IOB	Indirect object	RAD	Right adjunct
ATT	Attribute	IS	Independent structure
ADV	Adverbial	HED	Head

By reviewing 1050 financial reports, we found that attributes are often composed of compound words, such as "利润增速水平" (profit growth rate). Even attribute values are clause structures, such as "增长接近100%左右" (the growth is close to 100%). Most of the attribute values in the reports appear in the form of objects, and there are also ATT and COO dependencies among the constituent units. Similarly, some fixed dependencies exist between the constituent units of attributes, such as in the following forward-looking sentence: "预计公司的收入和毛利率将继续呈现高速增长的趋势" (It is expected that the company's revenue and gross profit margin will continue to show a trend of rapid growth). The results of the dependency syntactic analysis are shown in Figure 3.

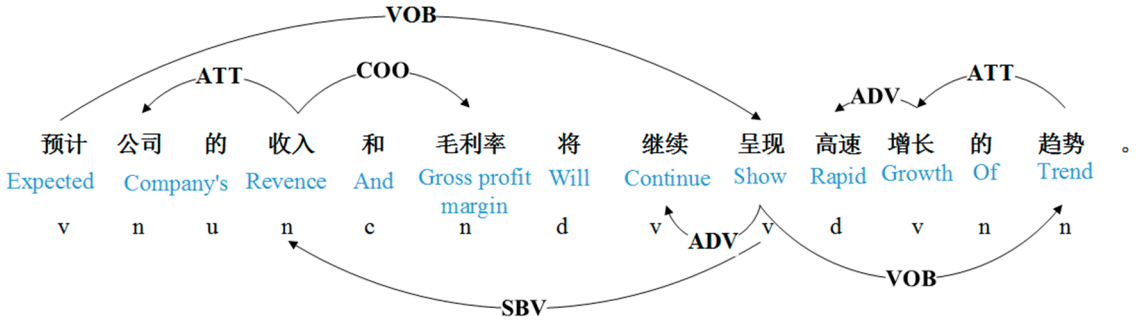

Figure 3. Dependency analysis results.

The attributes of this forward-looking sentence are "收入" (revenue) and "毛利率" (gross profit margin), and the syntactic role they play in a sentence is providing the subject for the verb "呈现" (show), namely, there is an SBV relationship. There is also a COO dependency between "收入" (revenue) and "毛利率" (gross profit margin). The attribute value element of the sentence is "呈现高速增长的趋势" (show a trend of rapid growth), which is part of the subordinate clause, where "呈现" (show) is the object of "预计" (expected), namely, a VOB relationship. Within the attribute value, there is an ADV relationship between "高速" (rapid) and "增长" (growth), a VOB relationship between "呈现" (show) and "趋势" (trend), and an ATT dependency relationship between "增长" (growth) and "趋势" (trend).

3.3.2. POS Syntactic Rules

As shown in Tables 2 and 3, certain POS syntactic rules are obtained using the context of attributes and attribute values via a corpus analysis. $Rule_{id}$ represents the encoding of rules; description represents the specific content of the rule; and output represents the recognized elements. where Ds is the dependency relation between word W_i and W_j (W_k) and Ds contains {VOB, ATT, ADV, COO, CMP, ... }, where W_j (W_k) is the first or last n words of W_i. Therefore, the dependency relation is based on the context of W_i. POS encompasses the part-of-speech information for W_i. Con (W_i~W_j) represents the connection from W_i to W_j. The arrows represent the dependency relations. For example, $W_i \rightarrow Ds \rightarrow W_j$ denotes that Wi depends on W_j through a syntactic relation Ds, where W_i is the father node of W. $R1_i$ indicates that an attribute (Attr) or attribute value (Val) is identified using only one dependency relation, and $R2_i$ indicates that there is more than one type of dependency relationship.

Table 2. POS syntactic rules based on the contextual rules of attributes.

$\mathbf{Rule_{id}}$	Description	Output
$R1_1$	$W_i \rightarrow D_S \rightarrow W_j$ s.t. $W_j \in$ {context (W_i)}, DS = ATT, POS (W_j, W_i) = {(n, n), (v, n), (nl, n)}	Con (W_j ~W_i)
$R1_2$	$W_i \rightarrow D_S \rightarrow W_j$ s.t. $W_j \in$ {context (W_i)}, D_S = COO, POS (W_i, W_j) = (n, n)	Con (W_i ~W_j)
$R2_1$	$W_i \rightarrow D_{S1} \rightarrow W_j$ and $W_i \rightarrow D_{S2} \rightarrow W_k$ s.t. W_j, $W_k \in$ {context (W_i)}, D_{S1} = ATT, D_{S2} = ATT, POS (W_k, W_j, W_i) = {(n, v, n), (n, n, n)}	Con (W_k~W_j~W_i)
$R2_2$	$W_i \rightarrow D_{S1} \rightarrow W_j$ and $W_j \rightarrow D_{S2} \rightarrow W_k$ s.t. W_j, $W_k \in$ {context (W_i)}, D_{S1} = ATT, D_{S2} = ATT, POS (W_j, W_i, W_k) = (n, n, n)	Con (W_j~W_i~W_k)
$R2_3$	$W_i \rightarrow D_{S1} \rightarrow W_j$ and $W_k \rightarrow D_{S2} \rightarrow W_i$ s.t. W_j, $W_k \in$ {context(W_i)}, D_{S1} = ADV, D_{S2} = ATT, POS (W_j, W_i, W_k) = (a, v, n)	Con (W_j~W_i~W_k)

Table 3. POS syntactic rules based on the contextual rules of attributes values.

$\mathbf{Rule_{id}}$	Description	Output
$R1_1$	$W_i \rightarrow D_S \rightarrow W_j$ s.t. $W_j \in$ {context (W_i)}, D_S = VOB, POS (W_i, W_j) = {(v, v), (v, m), (v, n)}	Con (W_i ~W_j)
$R1_2$	$W_i \rightarrow D_S \rightarrow W_j$ s.t. $W_j \in$ {context (W_i)}, D_S = ADV, POS (W_j, W_i) = {(d, v), (a, v), (d, v), (d, a)}	Con (W_j ~W_i)
$R1_3$	$W_i \rightarrow D_S \rightarrow W_j$ s.t. $W_j \in$ {context (W_i)}, D_S = ATT, POS (W_i, W_j) = {(a, n), (m, q), (m, m), (b, n)}	Con (W_i ~W_j)
$R2_1$	$W_i \rightarrow D_{S1} \rightarrow W_j \rightarrow D_{S2} \rightarrow W_k$ s.t. W_j, $W_k \in$ {context (W_i)}, D_{S1} = VOB, D_{S2} = ADV, POS (W_i, W_k, W_j) = (v, d, v)	Con (W_i~W_k~W_j)
$R2_2$	$W_i \rightarrow D_{S1} \rightarrow W_j$ and $W_j \rightarrow D_{S2} \rightarrow W_k$ s.t. W_j, $W_k \in$ {context (W_i)}, D_{S1} = VOB, D_{S2} = RAD, POS (W_i, W_j, W_k) = (v, m, m)	Con (W_i~W_j ~W_k)
$R2_3$	$W_i \rightarrow D_{S1} \rightarrow W_j$ and $W_j \rightarrow D_{S2} \rightarrow W_k$ s.t. W_j,$W_k \in$ {context(W_i)}, D_{S1} = VOB, D_{S2} = ATT, POS (W_i, W_k, W_j) = (v, b, n)	Con (W_i ~W_k ~W_j)
$R2_4$	$W_i \rightarrow D_{S1} \rightarrow W_j$ and $W_i \rightarrow D_{S2} \rightarrow W_k$ s.t. W_j,$W_k \in$ {context(W_i)}, D_{S1} = ADV, D_{S2} = CMP, POS (W_j, W_i, W_k) = (d, v, a)	Con (W_j ~W_i ~W_k)

3.3.3. Algorithm

Although a more complete integrity structure of elements can be obtained through the POS syntactic rules, the number of rules remains limited. The LSTM-CRF model can greatly improve the element recall rate, although the integrity is not guaranteed with this approach. Therefore, we can combine the advantages of both models and propose the LSTM-CRF model with the integrity algorithm. The method incorporates the advantages of the data-driven method and dependency syntax to raise the integrity and improve the accuracy of elements without losing the element recall rate.

The inputs of the LSTM-CRF model with the integrity algorithm include $\mathbf{LS}_{\text{tag}}$, $\mathbf{R}_{\text{tag}}$, and the following user specified parameters:

1. $\mathbf{LS}_{\text{tag}}$, the result set labeled by the LSTM-CRF model, including {B-Attr ("B-Attr" means the beginning of the attribute element), I-Attr ("I-Attr" means the reminder of attribute element), O ("O" means not belonging to any type of element)};
2. $\mathbf{R}_{\text{tag}}$, the result set labeled based on the POS syntactic rules, including {B-Attr, I-Attr, O};
3. **L**, the tags result set;
4. $\mathbf{R}_{\text{tag\_begin}}$, the subscript of the starting position of each attribute (B-Attr);
5. $\mathbf{R}_{\text{tag\_end}}$, the subscript of the ending position of each attribute (I-Attr);
6. flag, as a flag, its value is true or false;
7. *i*, as a variable to identify the position of each tag.

The main idea of the algorithm is to utilize the tags generated by the POS syntactic rules to correct the tags generated by the LSTM-CRF model. The algorithm can be divided into three steps. First, $\mathbf{R}_{tag}$ is traversed to find B-Attr, the current position is set as the superscript, and look for the subscript is obtained from the current label position [$\mathbf{R}_{tag\_begin}$: $\mathbf{R}_{tag\_end}$]. Second, $\mathbf{LS}_{tag}$ [$\mathbf{R}_{tag\_begin}$: R_{tag_end}] is traversed to determine whether the label of the interval only includes O and the label; if it does, then $\mathbf{LS}_{tag}$ is covered with $\mathbf{R}_{tag}$ in the interval, and otherwise, it will not be replaced. Third, the next set of annotations after $\mathbf{R}_{tag\_end}$ is searched until $\mathbf{R}_{tag}$ is traversed. By correcting the recognition strategy, the integrity of the elements will be guaranteed as much as possible. The LSTM-CRF model with integrity algorithm is shown in Algorithm 1.

Algorithm 1. LSTM-CRF model with integrity algorithm

Input: $\mathbf{LS}_{tag}$, $\mathbf{R}_{tag}$, flag, *i*
Output: **L**
Take the result of attribute tagging as an example
for (*i* = 0; *i* < $\mathbf{R}_{tag}$.length; *i*++)
    set $\mathbf{R}_{tag\_begin}$ = 0, $\mathbf{R}_{tag\_end}$ = 0
    if ($\mathbf{R}_{tag}$ [*i*]! = O)
        $\mathbf{R}_{tag\_begin}$ = *i*;
    while ($\mathbf{R}_{tag}$ [*i*]! = O)
        i++; $\mathbf{R}_{tag\_end}$ = *i*;
    flag = true;
    for ls ∈ $\mathbf{LS}_{tag}$ [$\mathbf{R}_{tag\_begin}$: $\mathbf{R}_{tag\_end}$]
    if ls! = O or ls! = $\mathbf{R}_{tag}$ [*i*]
        flag = false;
    if (flag)
        $\mathbf{LS}_{tag}$ [$\mathbf{R}_{tag\_begin}$: $\mathbf{R}_{tag\_end}$] = $\mathbf{R}_{tag}$ [$\mathbf{R}_{tag\_begin}$: $\mathbf{R}_{tag\_end}$]
        L← $\mathbf{LS}_{tag}$
return **L**

4. Experiment

This section evaluates the performance of our proposed method through experiments and a results analysis. The first part describes the source of the data, the second part introduces the evaluation indicators, and the third part describes the training settings of the LSTM-CRF model. Finally, two sets of experiments show that the LSTM-CRF model with the integrity algorithm provides better results compared with the LSTM-CRF model and POS syntactic rules and our proposed method has good domain independence.

4.1. Data Description

The object of this article's research is to determine whether elements in firm research reports can be recognized at the sentence level. However, Chinese firm reports tagging their corpus are not publicly available. Therefore, we have constructed a forward-looking information corpus of Chinese firm research reports. Financial websites are crawled to get the experimental data, the source data are extracted in html, and the crawled pages are denoised. The size of the corpus is shown in Table 4.

To standardize the data set, reports from different companies in different fields are used and students from financial fields are invited as annotators, with three students used to annotate the same data. The results are determined by the principle of the minority obeying the majority.

Table 4. Corpus size statistics, where En, Attr, and Val represents entity, attribute, and value, respectively.

Fields	Articles	FLSs	En	Attr	Val
Pharmaceutical industry	600	1362	1587	1887	3876
Medical industry	225	594	726	912	1344
Car manufacturer	225	197	1026	1011	1431
Total	1050	2547	3339	3810	6651

4.2. Evaluation Indicators

The recognition effectiveness can be evaluated based on the precision (P), recall (R), and F-score (F). To better assess the experimental results, accuracy and coverage evaluations were conducted.

Accuracy evaluations: The recognition results are compared to manual markings and are required to be fully compliant.

Coverage evaluation: This requires that the recognition result partially overlaps with the corpus of the tag. However, due to the different lengths of the constituent elements of the entities, attributes, and attribute values in this paper, the coverage evaluation cannot be unified.

In most cases, entities and attributes are formed by combining two nouns. The standard for coverage evaluation of entities and attributes is a partial overlap between the recognition results and manual markers. Denoted as "利润率增速" (Profit rate growth), the recognition results of "利润率" (Profit rate) and "增速" (growth) are considered to match.

The composition of the attribute value is longer than that of other elements, and partial matching can lead to unclear results. Therefore, the coverage evaluation of attribute values requires the recognition results to contain the manual markers, which is considered a correct match. For example, for the label "同比上升约2.5个百分点" (increased by about 2.5 percentage points year-on-year), "将同比上升约2.5个百分点" (will increase by about 2.5 percentage points year-on-year) is the correct match.

4.3. Experimental Settings

Our experiments are based on the Python language and the TensorFlow platform. We used LTP as the tool for word segmentation POS dependency syntax analysis. Experiments are implemented in a Python 3.6 version on CentOS 7.4 running on a server with a system configuration 16-core Intel Xeon E5 processor (2.10 GHz) with 16 GB RAM.

During the training of the network model, the pre-trained embedding with dimensions of 100 generated by the continuous bag-of-words (CBOW) model is directly used as the input of the LSTM-CRF model and used the Sogou news data (http://www.sogou.com/labs/resource/list_news.php) as the corpus to train the word embedding. To further improve the performance of the model and prevent overfitting, the dropout rate was fixed at 0.5 for all dropout layers in all experiments and the LSTM-CRF model was trained using backpropagation algorithms to optimize the parameters. In addition, the epoch was 20, and the batch size was 50. Stochastic gradient decent (SGD) algorithm was adopted with a learning rate of 0.05 for every epoch based on the training set.

4.4. Experimental Results and Analysis

To obtain reasonable and credible results, we use the corpus of the pharmaceutical industry to conduct five-fold crossover experiments. In this experiment, the experimental data are randomly divided into five subsets, of which four were used as training sets and one was used as the test set. This process was repeated five times.

4.4.1. The Results of the Five-Fold Crossover Experiments

In Table 5, we can see that the LSTM-CRF model performed better overall than the rules method because the LSTM has a strong sequence modeling advantage and the CRF can optimize the entire

sequence to make up for the local optimization problem of the LSTM. Therefore, the performance of the LSTM-CRF in the NER task was outstanding, which has been demonstrated by many previous studies. However, compared with the rules method, the precision indicator of the LSTM-CRF model was slightly lower because it is more driven by data and the text is converted to a vector at the time of input, which accelerates the data processing. However, this process also leads to the loss of many features of the language itself. Therefore, in the recognition results, we can find the identified boundaries of long elements, such as Attr and Val, were not sufficiently accurate; thus, the integrity of element recognition needs to be improved. The POS syntactic rules method presents lower recall and slightly higher precision because the rules of manual collection are always limited.

To incorporate the advantages of both, the LSTM-CRF model with the integrity algorithm (our method) is proposed, which can improve the recall rate under the data-driven approach and improve the precision under the syntactic dependence. Table 5 demonstrates that our method works better than the LSTM-CRF model and the POS syntactic rule method.

In Table 5, the coverage evaluation indexes of the LSTM-CRF model, POS syntactic rules, and our model are higher than that of the accuracy evaluation. Moreover, the precision of attribute value of the LSTM-CRF model (P = 85.99) was higher than that of the rules (82.83) under the coverage evaluation, indicating that the positioning of each element by the LSTM-CRF was relatively accurate while the positioning of element boundaries needs further improvement, which is the focus of this paper.

Table 5. Average the results of five-fold crossover experiments (%).

Elements	Model	Accurate Evaluation			Coverage Evaluation		
		P	**R**	**F**	**P**	**R**	**F**
En	LSTM-CRF	56.38	38.53	45.75	70.28	60.43	64.96
	Rules	-	-	-	-	-	-
	Our method	64.39	45.08	52.98	77.33	65.82	71.09
Attr	LSTM-CRF	71.35	58.67	64.19	81.79	70.57	75.69
	Rules	79.72	51.57	62.47	83.26	61.88	70.65
	Our method	81.86	63.59	71.47	86.46	74.17	79.74
Val	LSTM-CRF	76.81	72.61	74.62	85.99	77.31	81.38
	Rules	81.55	57.01	66.86	82.83	67.87	74.52
	Our method	86.09	73.62	79.34	90.17	81.84	85.64

From the perspective of the type of element, the indicators suggest that the entity recognition results (F = 45.75) were poor compared to those for the attributes (F = 64.19) and attribute values (F = 74.62), which can be explained as follows: first, entities in the economic field are more generalized and constantly accompanied by the emergence of new entities; and second, the position of the entity is usually adjacent to the attribute and the POS is consistent with the attribute, which is common in noun combinations. Therefore, even if the entity is recognized, there is a high probability that it may be recognized as an attribute, thus resulting in poor results. Obviously, the entity recognition rate is slightly improved under our method as shown in Figure 4 because the attribute recognition rate is improved, which effectively reduces the interference with entity recognition.

In Table 5, the attribute (F = 64.19) and the attribute value (F = 74.62) were overall optimistic under the LSTM-CRF model. Because the attributes and attribute values were mostly composed of phrases and clauses while the LSTM model has the characteristics of long-distance dependence, the LSTM-CRF model is suitable for the recognition of long elements. However, when using the POS syntactic rules method, the precision of attributes and attribute values (P = 79.72, P =8 1.55) was slightly higher than that of the LSTM-CRF (P = 71.35, 76.81) because we directly used the syntactic dependencies between words. The main idea of the integrity algorithm is to use the rule recognition results to correct the recognition boundary of the LSTM-CRF model and further improve the performance, which can be clearly observed in Figures 5 and 6.

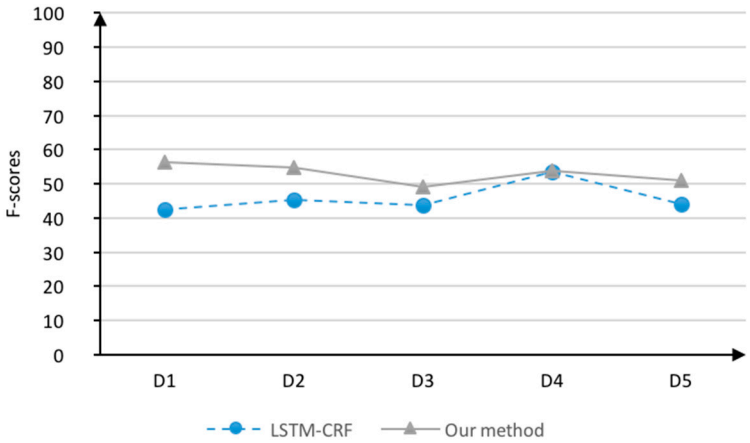

Figure 4. Comparison of F-scores of entity recognition in the five data sets.

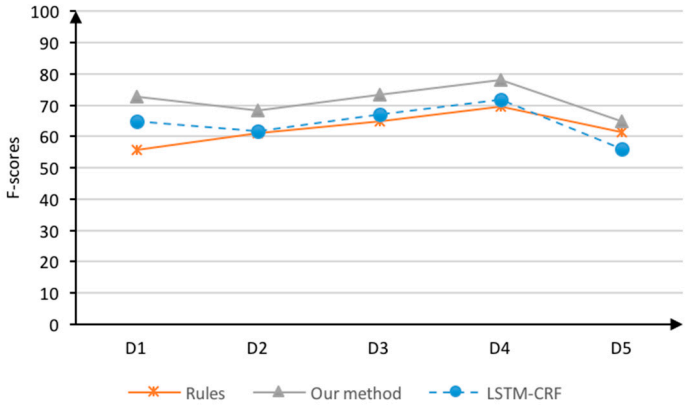

Figure 5. Comparison of the F-scores of attributes recognition in the five data sets.

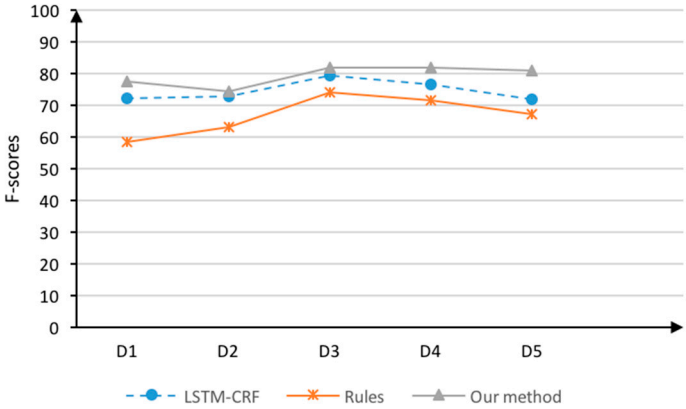

Figure 6. Comparison of the F-scores of attribute value recognition in the five data sets.

4.4.2. The Field Cross-Recognition Results

To further verify the effectiveness of the LSTM-CRF model with the integrity algorithm and determine whether the algorithm displays good domain independence, the corpus of the pharmaceutical industry is used as a training set. The corpus of the medical industry, which is similar to that for the pharmaceutical industry, is used as test set 1, and the corpus of the car manufacturing industry, which is not associated with the pharmaceutical industry, is selected as test set 2. The results are shown in Tables 6–8, respectively.

A comparison of the recognition results of test set 1 and test set 2 in Table 8 show that the F-scores of the test set 1 entities and attributes were higher than that of the test set 2 under the accurate evaluation, whereas the F-scores of the attribute values were close to each other, although test set 2

(F = 78.77) was slightly higher than test set 1 (F = 77.67). Entities and attributes were domain-related. An entity is a unique name in a domain, and an attribute is the characteristics of the entity; therefore, the test set 1 entity and attribute recognition performances were slightly better. The attribute value was not relevant to the domain and represents the scope of an attribute, and its constituent elements often have similarities. Therefore, the recognition of results for the attribute values of test set 1 and test set 2 were close to each other, and test set 2 may even be slightly higher than test set 1.

A comparison of the Avg (average value) of Table 8 with the Avg of each element in our method in Table 5 shows that the recognition results of our method were indeed higher than that shown in Table 5 because the LSTM-CRF model had field adaptability. Although the two sets of experiments prove that the cross-domain recognition performance was not as good as the training performance in the same field, the overall result shows that the recognition effect was not much different, which indicates that our method has a certain degree of versatility.

Finally, a comparison of the results in Tables 6–8 shows that the F-scores of the entities, attributes, and attribute values in our model are much higher than that of the LSTM-CRF model and POS syntactic rules. This finding again shows that our method recognizes phrase elements and clause-level elements and thus can effectively improve the integrity of the elements. In addition, the experiment proves that our method also has advantages in other fields, which fully demonstrates that the model is domain independent.

Table 6. Field cross-recognition results for the LSTM-CRF model (%).

Test Set	Elements	Accurate Evaluation			Coverage Evaluation		
		P	R	F	P	R	F
1	En	56.25	36.00	43.90	65.35	58.06	61.49
	Attr	70.15	46.53	55.95	77.34	54.87	64.20
	Val	71.83	71.83	71.83	80.82	76.39	78.20
2	En	56.92	43.79	49.50	63.32	51.80	56.94
	Attr	59.58	41.94	49.23	73.25	48.62	58.45
	Val	70.25	67.82	69.01	83.18	76.03	79.44
Avg	En	56.59	39.90	46.70	64.33	54.93	59.22
	Attr	64.87	44.24	52.59	75.30	51.75	61.33
	Val	71.04	69.83	70.42	82.00	76.21	78.82

Table 7. Field cross-recognition results for the rules (%).

Test Set	Elements	Accurate Evaluation			Coverage Evaluation		
		P	R	F	P	R	F
1	Attr	74.28	40.13	52.11	75.52	43.37	55.10
	Val	78.60	53.74	63.84	82.48	60.44	69.76
2	Attr	63.21	37.86	47.36	70.14	44.90	54.75
	Val	72.93	50.54	59.70	77.36	53.02	62.92
Avg	Attr	68.75	39.00	49.74	72.83	44.14	54.93
	Val	75.77	52.14	61.77	79.92	56.73	66.34

Table 8. Field cross-recognition results for the Our method (%).

Test Set	Elements	Accurate Evaluation			Coverage Evaluation		
		P	R	F	P	R	F
1	En	57.35	41.20	47.95	65.21	50.82	57.12
	Attr	80.62	58.23	67.62	85.14	67.44	75.26
	Val	83.10	72.91	77.67	88.62	75.23	81.38
2	En	53.37	39.50	45.40	61.67	45.20	52.17
	Attr	69.93	43.46	53.61	79.95	61.74	69.67
	Val	78.03	79.52.	78.77	84.29	80.59	82.40
Avg	En	55.36	40.35	46.68	63.44	48.01	54.65
	Attr	75.28	50.85	60.62	82.55	64.59	72.47
	Val	80.57	76.25	78.22	86.46	77.91	81.89

4.4.3. Comparative Experiments

To further validate the effectiveness of our approach, we specifically compare our methods with several advanced methods for attribute. The test results are as seen in Table 9.

Table 9. Comparative experiments of attribute (%). The F value is calculated under the precise evaluation.

Model	F
CRF	60.65
LSTM-CRF	64.19
Char-LSTM-CRF	72.56
LSTM-CNNs-CRF	73.09
Our method	72.87
LSTM-CNNs-CRF + Integrity algorithm	75.21

CRF [23]: The CRF model was proposed by Lafferty. It incorporates the word, POS, n-gram words, suffix and prefix, and location features.

LSTM-CRF [16]: In article, the LSTM-CRF model is like that proposed by Huang et al. [16].

Char-LSTM-CRF [18]: The character embedding of the words is given to a bidirectional LSTM. Finally, it concatenates outputs to an embedding from a lookup table to obtain a representation for this word.

LSTM-CNNs-CRF [24]: The CNN training obtains the character embedding, then connects the character embedding and the word embedding, and inputs it into the LSTM. Finally, it inputs the vector output from the LSTM into the CRF to jointly decode to obtain the optimal sequence label.

As can be seen from Table 9, our method is significantly superior to the basic methods of CRF and LSTM-CRF. Compared to the Char-LSTM-CRF and LSTM-CNNs-CRF models, we consider syntactic features, which works well. In addition, we can also see that combining LSTM-CNNs-CRF with integrity algorithm can achieve good results. This is because the integrity algorithm can analyze semantics from a language perspective and improve the integrity of element extraction. This again illustrates the validity and migration of the integrity algorithm.

5. Conclusions

In summary, our research focuses on firm reports in the financial field. FLSs are used as objects to recognize valuable financial information, such as entities, attributes, and attribute values. Considering the three different types of elements, a synchronous recognition strategy with the advantages of dependency syntax is incorporated to capture the structure of elements and define POS syntactic rules based on the contexts of attributes and attribute values. Then, the integrity algorithm is used to correct the boundaries of the LSTM-CRF model labeling results. Finally, without losing

the recall rate, the accuracy of the model is improved by correcting the integrity of the element, thereby optimizing the model performance. In addition, experiments in different fields are repeated. The experiments showed that the proposed model displays good domain independence and can be easily applied in various fields. Integrity algorithms can also be easily combined with neural network models to avoid relying solely on being data driven.

The next steps in this research are as follows. First, we will continue to conduct research on the boundaries of elements and further improve the effectiveness of recognition. Second, because information on the Internet is both genuine and fake, methods of distinguishing between genuine and fake information and selecting high-value information should be investigated. Third, determining how to use the identified elements to interpret the current status of a company and provide decision support and early warnings will be a focus of upcoming research.

Author Contributions: Methodology, experimental analysis, and paper writing, R.G.; The work was done under the supervision and guidance of D.X. and Z.N.

Funding: This work is sponsored by Natural Science Foundation of Shanghai, Project Number: 16ZR1411200.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ding, X.; Zhang, Y.; Liu, T.; Duan, J. Deep learning for event-driven stock prediction. In Proceedings of the International Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015; pp. 2327–2333.
2. Lee, H.; Surdeanu, M.; MacCartney, B.; Jurafsky, D. On the importance of text analysis for stock price prediction. In Proceedings of the LREC 2014, Reykjavik, Iceland, 26–31 May 2014; pp. 1170–1175.
3. Qiu, X.Y.; Srinivasan, P.; Hu, Y. Supervised learning models to predict firm performance with annual reports: An empirical study. *J. Assoc. Inf. Sci. Technol.* **2014**, *65*, 400–413. [CrossRef]
4. Kumar, A.; Sethi, A.; Akhtar, M.S.; Ekbal, A.; Biemann, C.; Bhattacharyya, P. IITPB at SemEval-2017 Task 5: Sentiment Prediction in Financial Text. In Proceedings of the International Workshop on Semantic Evaluation, Vancouver, BC, Canada, 3–4 August 2017; pp. 894–898.
5. Kumar, A.; Alam, H.; Werner, T.; Vyas, M. Experiments in Candidate Phrase Selection for Financial Named Entity Extraction-A Demo. In Proceedings of the 26th International Conference on Computational Linguistics: System Demonstrations, Osaka, Japan, 11–16 December 2016; pp. 45–48.
6. Jiang, T.Q.; Wan, C.X.; Liu, D.X. Evaluation Object-Emotional Word Pair Extraction Based on Semantic Analysis. *Chin. J. Comput.* **2017**, *40*, 617–633.
7. Li, F. The information content of forward-looking statements in corporate filings—A naïve Bayesian machine learning approach. *J. Account. Res.* **2010**, *48*, 1049–1102. [CrossRef]
8. Feldman, R.; Fresco, M.; Goldenberg, J.; Netzer, O.; Ungar, L. Extracting product comparisons from discussion boards. In Proceedings of the IEEE International Conference on Data Mining, Omaha, NE, USA, 28–31 October 2007; pp. 469–474.
9. Hu, M.; Liu, B. Mining and summarizing customer reviews. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WA, USA, 22–25 August 2004; pp. 168–177.
10. Qiu, G.; Liu, B.; Bu, J.; Chen, C. Opinion word expansion and target extraction through double propagation. *Comput. Linguist.* **2011**, *37*, 9–27. [CrossRef]
11. Chieu, H.L.; Ng, H.T. A maximum entropy approach to information extraction from semi-structured and free text. In Proceedings of the Eighteenth National Conference on Artificial Intelligence, Edmonton, AB, Canada, 28 July–1 August 2002; pp. 786–791.
12. Zhang, S.; Jia, W.J.; Xia, Y.; Meng, Y.; Yu, H. Extracting Product Features and Sentiments from Chinese Customer Reviews. In Proceedings of the 7th LREC, Valletta, Malta, 17–23 May 2010; pp. 17–23.
13. Finkel, J.R.; Grenager, T.; Manning, C. Incorporating non-local information into information extraction systems by Gibbs sampling. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, Ann Arbor, MI, USA, 25–30 June 2005; pp. 363–370.

14. Choi, Y.; Cardie, C.; Riloff, E.; Patwardhan, S. Identifying sources of opinions with conditional random fields and extraction patterns. In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, BC, Canada, 6–8 October 2005; pp. 355–362.
15. Pinheiro, P.H.; Collobert, R. Recurrent convolutional neural networks for scene parsing. *arXiv*, **2013**, arXiv:1306.2795.
16. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF models for sequence tagging. *arXiv*, **2015**, arXiv:1508.01991.
17. Limsopatham, N.; Collier, N.H. Bidirectional LSTM for named entity recognition in Twitter messages. In Proceedings of the 26th International Conference on Computational Linguistics, Osaka, Japan, 11–16 December 2016; pp. 145–152.
18. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural architectures for named entity recognition. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 260–270.
19. Popescu, A.M.; Etzioni, O. Extracting product features and opinions from reviews. In *Natural Language Processing and Text Mining*; Springer: London, UK, 2007; pp. 9–28.
20. Bloom, K.; Garg, N.; Argamon, S. Extracting appraisal expressions. In Proceedings of the Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics, Rochester, NY, USA, 22–27 April 2007; pp. 308–315.
21. Somprasertsri, G.; Lalitrojwong, P. Mining Feature-Opinion in Online Customer Reviews for Opinion Summarization. *J. Univ. Comput. Sci.* **2010**, *16*, 938–955.
22. Graves, A. *Supervised Sequence Labelling with Recurrent Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2012; ISBN 9783642212703.
23. Lafferty, J.; McCallum, A.; Pereira, F.C. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of the Eighteenth International Conference on Machine Learning Morgan Kaufmann Publishers, San Francisco, CA, USA, 28 June–1 July 2001; pp. 282–289.
24. Ma, X.; Hovy, E. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. *arXiv*, **2016**, arXiv:1603.01354.