Who Blames Whom in a Crisis? Detecting Blame Ties from News Articles Using Neural Networks

Shuailong Liang, Olivia Nicol, Yue Zhang
Singapore University of Technology and Design
8 Somapah Road, Singapore 487372
shuailong_liang@mymail.sutd.edu.sg, {olivia_nicol, yue_zhang}@sutd.edu.sg

Abstract

Blame games tend to follow major disruptions, be they financial crises, natural disasters or terrorist attacks. To study how the blame game evolves and shapes the dominant crisis narratives is of great significance, as sense-making processes can affect regulatory outcomes, social hierarchies, and cultural norms. However, it takes tremendous time and efforts for social scientists to manually examine each relevant news article and extract the blame ties (A blames B). In this study, we define a new task, Blame Tie Extraction, and construct a new dataset related to the United States financial crisis (2007-2010) from *The New York Times*, *The Wall Street Journal* and *USA Today*. We build a Bi-directional Long Short-Term Memory (BiLSTM) network for contexts where the entities appear in and it learns to automatically extract such blame ties at the document level. Leveraging the large unsupervised model such as GloVe and ELMo, our best model achieves an F1 score of 70% on the test set for blame tie extraction, making it a useful tool for social scientists to extract blame ties more efficiently.

Introduction

Blame is an issue that has been receiving increasing attention in the social sciences in recent years (Alicke 2000; Hobolt and Tilley 2014; Hood 2010; Shaver 2012). In particular, more attention has been placed on blame dynamics following major disruptions, such as natural disasters (Malhotra and Kuo 2008), financial crises (Nicol 2016; Tourish and Hargie 2012), and terrorist attacks (Olmeda 2008). Studying blame is of great significance as sense-making processes inform what and who a society values, and ultimately shape lawmaking. For instance, the intense blame targeting Wall Street during the financial crisis (2007-2010) helped lawmakers pass the July 2011 Dodd-Frank Wall Street Reform and Consumer Protection Act.

Although the problem is important, it takes tremendous time and efforts for social scientists to manually examine each relevant news article and extract the blame ties (A blames B). In Figure 1, for example, the tuple (John B. Taylor, Fed) is extracted as a blame tie. Recently, deep neural networks have proved very powerful at solving many social science problems (Li and Hovy 2014; Rule, Cointet, and Bearman 2015; Bail 2016). Based on the dataset annotated from several new media excerpts on blame, we investigate automatic ways to extract blame ties from new articles.

> **John B. Taylor**, a professor of economics at Stanford University and a former under secretary of the Treasury under President Bush, recently argued that the **Fed**'s rate cuts after 2001 appeared to have *exaggerated both the housing boom and bust*

Figure 1: An example sentence from our dataset containing a blame tie. The red/**bold** words are entities involved in a blame tie, and the blue/*italic* words are supporting evidence that the blame tie exists.

There are three main challenges. First, some patterns only have blame meanings in specific contexts. For instance, *"It was lenders that made the lenient loans, it was home buyers who sought out easy mortgages, and it was Wall Street underwriters that turned them into securities."* (*The Wall Street Journal*, Aug 2007), only with the background of the financial crisis can we identify that the blamed targets are *lenders*, *home buyers* and *Wall Street*. Second, there are many ways to attribute blame and the structure of the sentences can be quite complex. Third, it is common for journalists to use metaphors and ironies to designate actors.

We design several neural models to address the problem. First, we leverage a neural network to learn the prior information about entities for blame tie extraction. In particular, a neural network is used to learn dense vector representations of entities so that similar entities can be visualized close to each other in the embedding space, and the likeliness of one entity to blame another can be inferred automatically without further knowledge. Second, we build a BiLSTM neural network to represent contexts, which can be used to predict blame ties between entities mentioned in the news articles using linguistic clues. Finally, a model that integrates entity knowledge and linguistic knowledge is constructed by integrating the two respective networks.

We conduct a case study on blame games for the U.S. financial crisis (2007-2010), the most important event since the Great Depression, which led to at least $6 trillion in losses (Luttrell et al. 2013). Results show that our model can

Copyright © 2019, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Articles
... Ordinarily, **[Americans]** welcome lower interest rates. But many feel differently this time. Some think the economy is fine and inflation is the main danger. But a moral element is also at work: Many think a rate cut would reward foolish speculation and **[Wall Street]** greed at the expense of the thrifty (1). “The **[Federal Reserve]** needs to stand its ground and not bail out hedge funds – they should have known better to begin with!” **[Suzanne Mitchell]**, an administrative assistant at a Houston real-estate company, says in an email (2). In an interview, she adds: “I’m very sorry that **[people]** took out $450,000 mortgages with no money down ... **[people]** ought to be responsible for the loans they take out.” (3) . . . But Mr. **[Brason]** contrasts that with the far greater reliance on borrowed money that is typical nowadays. Some “of us ... weren’t buying up five or 10 properties without any money down,” he says. “**[People]** took the risks and should pay the price. (4) A lot of others at the higher end of the food chain, the **[investment bankers]** and **[hedge-fund managers]** were making oodles of fee-income money and frankly, there’s a lot of public opinion that it was excessive. (5) ” . . .

Blame Source	**Blame Target**	**Causality Link**
Americans(e_1)	Wall Street(e_2)	(1)
Suzanne Mitchell(e_3)	Federal Reserve(e_4)	(2)
Suzanne Mitchell(e_3)	people(e_5)	(3)
Brason(e_6)	people(e_5)	(4)
Brason(e_6)	investment bankers (e_7)	(5)
Brason(e_6)	hedge-fund managers (e_8)	(5)

Table 1: An article titled *Rate Cut Has Foes on Main Street* (*The Wall Street Journal*, September 2007). Top: paragraphs of the article containing several blame patterns. The entities are in the brackets. Bottom: blame ties extracted from the article.

target **source**	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
e_1	-	1	0	0	0	0	0	0
e_2	0	-	0	0	0	0	0	0
e_3	0	0	-	1	1	0	0	0
e_4	0	0	0	-	0	0	0	0
e_5	0	0	0	0	-	0	0	0
e_6	0	0	0	0	1	-	1	1
e_7	0	0	0	0	0	0	0	0
e_8	0	0	0	0	0	0	0	0

Table 2: Matrix representation of the blame ties in Table 1.

effectively learn both entity knowledge and linguistic clues for blame ties. For example, it can successfully extraction entity relation with regard to blame ties from the crisis, such as the fact that both Wall Street and McCain tend to blame the same targets (Obama and Bernanke). In addition, the model can generalize to new cases of extracting blame patterns automatically. Our implementation and trained models are released at https://github.com/Shuailong/BlamePipeline.

Related Work

NLP has become increasingly popular in the social science area. O’Connor et al. [2010] aligned sentiment measured from Twitter to public opinion measured from polls, and found the two correlate well. Bamman and Smith [2015] tried to use text data to estimate the political ideologies of individuals. Mohammad et al. [2016] create the SemEval 2016 Task 6 called Stance Detection Task, which detects the Twitter user’s stance towards a target of interest. Preoţiuc-Pietro et al. [2017] also predicted the political ideologies of Twitter users, in a more fine-grained form. Social scientists used NLP along with network analysis etc. to analyze social media texts (Rule, Cointet, and Bearman 2015) and the State of Union addresses in United States (Bail 2016).

The Blame Tie Extraction task can be regarded as a special case of relation extraction (Miwa and Bansal 2016). Relation extraction solves the task of classifying a pair of entities into one of several pre-defined categories, such as Cause-Effect and Component-Whole (Hendrickx et al. 2009), while the Blame Tie Extraction task requires extracting all the blame ties among the entities of interest in an article. Our work differs from existing work on relation extraction in two main aspects. First, our work is at the document level and the data is sparse, while most existing work on relation extraction focuses on the sentence level (Nguyen and Grishman 2015). Second, our work explicitly uses entity prior information on blame patterns, which does not make sense in general domain relation extraction. Most existing work mixes entity and content information in modeling relations.

In blame game research, social scientists care more about a few key players instead of all the entities, and most entities in the passage are irrelevant for studying the blame game (Nicol 2016). Therefore, in this paper, we assume that the entities of interest are already given, and we only need to extract blame ties from these entities.

To the best of our knowledge, our work is the first to study the Blame Tie Extraction task using NLP techniques.

Dataset

We manually create a dataset on the U.S. financial crisis. The dataset is drawn from three newspapers in the U.S., including *The New York Times*, *The Wall Street Journal* and *USA Today*, chosen for three main reasons. First, they are the most widely circulated newspapers in the United States. Second, they cover the social spectrum from elite to mass. Third, they also cover the political spectrum: from the quite liberal *New York Times*, to the centrist *USA Today*, to the conservative *Wall Street Journal* (Gentzkow and Shapiro 2010; Groseclose and Milyo 2005). The time period studied here

	USA	**NYT**	**WSJ**
days	310	736	648
articles	132	429	438
blame ties	353	787	754

Table 3: Dataset size for the three newspapers. USA: *USA Today*. NYT: *The New York Times*. WSJ: *The Wall Street Journal*.

spans from August 2007, after the first warning signs for the crisis appeared, to June 2010, right before the signature of the largest set of financial regulations since the Great Depression.

We use a set of keywords to filter the articles gathered from Factiva[1] and LexisNexis[2], getting articles containing blame patterns for the crisis. The keywords we use are blame-related (*attack*, *accuse*, *misconduct*, ...) or event-related (*financial crisis*, *global recession*, *housing bubble*, ...)[3]. There are in total 70 blame-related keywords and 13 event-related keywords (stem form). The two classes of keywords are combined together to filter the articles.

Blame incidences are manually coded for each article. The identification of a blame pattern requires the presence of 1) a blame source, 2) a blame target, and 3) a causality link. An example of a sentence containing a blame instance would be as follows: "*Sen. Richard Shelby, R-Ala. [Blame Source]... said the FED [Blame Target] 'kept interest rates too low for too long, encouraging a housing bubble and excessive risk taking'[causality link].*" (*USA Today*, December 2009). The blame source and target form a blame tie. Table 1 gives an example article and its annotations and Table 3 shows the statistics of the dataset with the number of blame ties.

To ensure the reliability of the dataset, we ask two more annotators to annotate a subset of the dataset. Specifically, we sample 100 articles, including 13 articles from *USA Today*, 43 articles from *The New York Times*, and 44 articles from *The Wall Street Journal*, in proportion to the respective number of articles of the three newspapers in the whole dataset. Then we run evaluations using the two annotators' results against the gold data. The average of the F1 score of the two annotators is 94.425%, and the Fleiss's kappa is 0.8744, which illustrates the strong inter-annotator agreement of the dataset.

In the training process, the annotated blame ties serve as positive samples. The negative samples are generated by removing the positive entity pairs from all possible permutations of the entities of interest. The sample statistics about the whole dataset are shown in Table 4. Theoretically, the number of negative samples increases quadratically with the number of entities in the article. In our dataset, the average number of entities we consider per article is 3. Therefore the dataset is rather balanced.

number of articles	998
number of samples	8562
number of entities/article	2.97
average neg/pos ratio per article	2.19
total neg/pos ratio	3.61

Table 4: Sample statistics.

Task

We formulate the Blame Tie Extraction task as follows. Given a news article d and a set of entities e, we have $|e| \cdot (|e| - 1)$ possible directed links among them. We assign label 1 to a pair (s, t) when entity s blames entity t based on article d, otherwise we assign label 0 to the pair. We can use a matrix for a more intuitive illustration. For the example in Table 1, the matrix constructed is shown in Table 2.

For a given entity pair (s, t) with label l, we would like to maximize the likelihood

$$L = P(l|s, t, d), l \in \{0, 1\}$$

In order to predict whether a blame tie exists between two entities based on the article, we have two sources of information to utilize. One is the entities themselves. For instance, we know that Democrats tend to blame Republicans so as to weaken their political opponents, and tend to blame Wall Street so as to gain popular support to impose a stringent set of financial regulations. The other one is the contexts in which the entities are mentioned. We rely on linguistic patterns of sentences to extract the blame ties. For instance, in this sentence "*Who is to blame? Hedge funds, for one, he says.*" (*The Wall Street Journal*, Sept 2007), the linguistic structure indicates that the entity appearing after the question is the blame target entity, and the narrator is the blame source entity.

Models

We first introduce a simple rule-based model. Next, we introduce three neural models. The Entity Prior Model which directly extracts prior information about entities (i.e. which entities tend to blame which entities). The Entity Prior Model can learn entity information from its blame patterns such as political standing, and it can generalize to new events with the same entities. Then, we mask out the entity mentions from the text, relying purely on the contexts surrounding the mentions of the entities. The Context Model can generalize to new entities and across historical periods. Finally, a model combining the two is built to investigate the interactions.

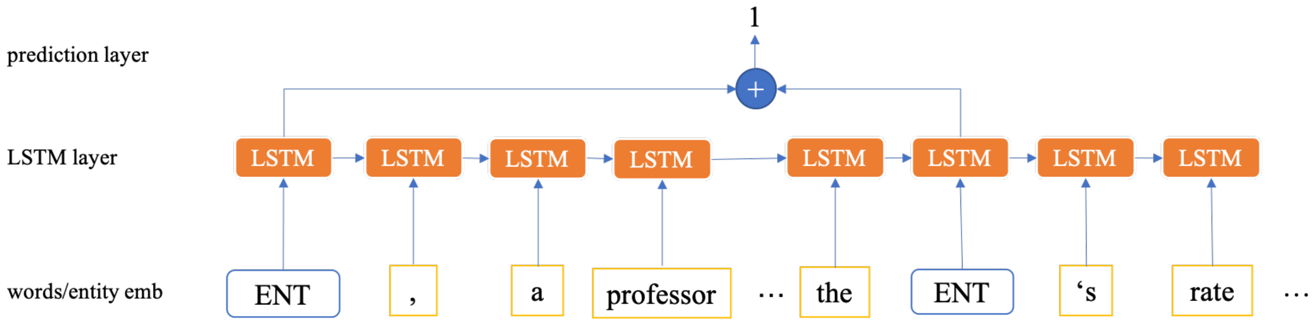
Rule-based Model

As a baseline model, we use a simple rule to decide if a blame tie exists between two entities: if the minimal sentence distance between the two entities are less than or equal to d, AND a blame related keyword appears in any of the sentences mentioning either of the two entities, we determine that a blame tie exists between the two entities. According to the distribution of the minimal distance between

[1] https://www.dowjones.com/products/factiva/

[2] https://www.lexisnexis.com

[3] The full list of keywords are released along with the code.



John B. Taylor, a professor of economics at Stanford University and a former under secretary of the Treasury under President Bush, recently argued that the **Fed**'s rate cuts after 2001 appeared to have *exaggerated both the housing boom and bust*.

Figure 2: Context Model. LSTM is used to encode the sentences, and the hidden vectors at the positions of the entity are pooled together into a single vector to represent the context of the entity. The source entity context vector and the target entity context vector are concatenated together to be sent to the prediction layer. The prediction layer predicts 1 if the source to target blame tie exists otherwise 0.

two entities in a blame tie in our dataset, over 90% of the blame entity pairs has a minimal sentence distance less than or equal to 3. Therefore, we set $d = 3$.

To determine the direction (A blames B or B blames A), we define the **aggressiveness** of the entity, which is the percentage of the entity being the blame source among all the blame ties related to this entity in the training data. The entity with higher aggressiveness is the blame source entity, and the other one is the blame target. When there is a tie, we use random guess. When the entity is unknown, we use 0.5 as the aggressiveness score.

Entity Prior Model

We use the fully connected feedforward neural network (FCN) to collect entity prior knowledge, namely who is likely to blame whom without additional information. To elaborate, we represent entities by their embeddings, concatenate the embeddings of the blame source and target entity, and then stack a fully connected layer to learn the interactions between the entities. The FCN outputs the probability of the blame tie between the source entity and the target entity:

$$f_{score} = ([\mathbf{E}^e(e_s); \mathbf{E}^e(e_t)]) \cdot W^e + b^e$$

e_s and e_t represent the source entity and target entity index, respectively, $\mathbf{E}^e$ is the embedding matrix for entities, ‘;’ is the concatenation operator, and W^e and b^e are parameters. Specifically, $\mathbf{E}^e \in \mathbf{R}^{n \times m}$, $n = 707$ is the number of entities in training set, m is the entity embedding dimension, which is tuned as hyperparameters. $W^e \in \mathbf{R}^{2m \times 2}$ and $b^e \in \mathbf{R}^2$. $\mathbf{E}^e$ is initialized with standard normal distribution. At test time, we use the ⟨UNK_ENT⟩ to indicate unknown entities from the training set.

Like word embedding, we hope to learn meaningful representations of the entities, i.e., the entities sharing similarity blame behavior patterns will stay close in the embedding space.

Context Model

The Entity Prior Model tells how likely a particular entity is to blame or be blamed without further information. In contrast, the Context Model relies only on linguistic clues from news articles, finding blame patterns explicitly or implicitly mentioned. The Context Model is thus useful across different political settings where the entities are different.

To model context information about an entity, we first locate the positions of all occurrences of the entity in the article. A position is represented by a tuple (i, j), where i and j denotes sentence number and word number, and the positions of blame source and blame target are denoted as pos^s and pos^t, respectively. Second, we replace each entity mention by a special ⟨ENT⟩ token, so that we do not have any information regarding the entity itself. Third, we run a bidirectional LSTM on sentences containing the entity and use the LSTM output to represent the context of each word:

$$h^i_j = \text{LSTM}(\mathbf{E}^w(w^i_j))$$

w^i_j is the j-th word of the i-th sentence, $\mathbf{E}^w$ is the embedding matrix for words, and h^i_j is the concatenation of the LSTM outputs of the last layers from both directions. If $(i, j) \in pos^s$ or $(i, j) \in pos^t$, h^i_j will be used to represent the context of entity s or t.

Since an entity may appear at multiple positions in one article, we may have multiple representations of the entity context. Pooling is used to reduce the embeddings into one single vector. The pooling result of the contexts representations of source and target entity are denoted as V_s and V_t, respectively:

$$V_e = \underset{(i,j) \in pos^e}{\text{pool}} (h^i_j), e \in \{s, t\}$$

	min	avg	max
sentences per doc	4	45	384
words per sentence	1	26	159
words per doc	133	1,209	9,064

Table 5: Sentence and words statistics.

where pool denotes the pooling function, for which we try random selection, mean, max and attention. For attention pooling, we use a two-layer feedforward neural network to score each vector representations, using softmax function to normalize them as weights, and use the weighted sum of the representations as the entity representation. Once we obtain representations of both the source and target entity, we concatenate them and then use a fully connected layer to learn a score of this entity pair:

$$f_{score} = ([V_s; V_t]) \cdot W^v + b^v$$

$W^v \in \mathbf{R}^{2H \times 2}$ and $b^v \in \mathbf{R}^2$ are both parameters. H is the dimension of BiLSTM outputs. The model architecture is depicted in Figure 2.

Combined Model

To incorporate the prior information about entities, we concatenate two additional vectors on the basis of the Context Model, which are the embeddings of the blame source entity $\mathbf{E}^e(s)$ and blame target entity $\mathbf{E}^e(t)$ from section :

$$f_{score} = ([\mathbf{E}^e(e_s); V_s; \mathbf{E}^e(e_t); V_t]) \cdot W^c + b^c$$

$W^c \in \mathbf{R}^{2H+2m}$ and $b^c \in \mathbf{R}^2$ are both parameters. We expect the model to simultaneously learn how to extract blame ties from context, and also learn the representations of the entities themselves as a byproduct.

Experiments

We conduct experiments on our dataset using the Entity Prior Model, the Context Model, and the Combined Model and compare the performance among different models. To find the best model architecture, we also conduct development experiments to investigate the effects of different pooling functions and pretrained word embeddings.

Experimental Settings

Stanford CoreNLP (Manning et al. 2014) is used to tokenize articles into sentences and words. The sentence length statistics of the dataset articles are shown in Table 5. There is no further preprocessing except that words are converted to lower case. The dataset is split into train, dev, and test set at the document level by the ratio of 8:1:1. We use F1 on the positive class to measure the performance of the model.

To investigate whether the learned entities representations are helpful in blame tie extraction, we conduct another round of evaluations on the known entities of dev and test set as shown in Table 8. KNOWN denotes entity pairs both of which appear in the training data, while ALL denotes all entity pairs. By comparing the model performance on KNOWN with that of ALL, we can evaluate the usefulness of entities representations. Conversely, we can also evaluate the robustness of the model against unknown entities.

Leveraging large unsupervised data has proved to be helpful for many NLP tasks, especially for tasks with small datasets. Word2Vec (Mikolov et al. 2013) or GloVe (Pennington, Socher, and Manning 2014) can be used to pretrain word embeddings on larger external dataset. ELMo (Peters et al. 2018) word vectors are internal states of a deep bidirectional language models, and can effectively capture the syntax and semantics of words. We conduct experiments on GloVe and ELMo, and investigate the effects of these pretrained word embeddings.

Our models are implemented in PyTorch[4]. During training, Adam (Kingma and Ba 2014) is used as the optimizer, and the default learning rate is adopted. We use the minibatch size of 50 for all three models. Dropout (Hinton et al. 2012) of 0.5 are used to prevent overfitting. Dropout is applied to word embeddings and RNN outputs. We set gradient clipping to 3 to stabilize the training process. The maximum number of epochs is 30 and we use early stopping techniques with a patience of 10 epochs. For the model hyperparameters, word embedding size is 100, LSTM hidden size is 100 for each direction, and entity embedding size is 50.

Development Experiments

Before turning to the BiLSTM model, we use the small window before and after the entity to extract the context information. Formally, we assume that one entity appears only once in a sentence. Given an entity e appearing at sentence s, if the index of e is i, and window size is w, we use concatenation of the embeddings of words $s_{i-w} \dots s_{i-1}$ and $s_{i+1} \dots s_{i+w}$ as the context of e. We call this model Bag of Embeddings (BoE). We try the window sizes of 3 and 6 and report the higher dev result. If an entity appears multiple times in an article, we randomly select one representation (BoERand).

As stated in the Context Model section, we compare different pooling methods used to aggregate multiple contexts representations, and bi-directional LSTM with a single forward LSTM. The results of ALL dataset are shown in Table 6. BiLSTM works better than LSTM for every pooling method since we can take advantage of information from both directions of the sentence. Random pooling has the worst result, mainly because a lot of important information is lost. Max pooling has the highest score on most cases, therefore we use BiLSTM + max pooling in the following experiments.

In comparison to BoE models, LSTM models are worse. The reason may be that BoE models consider contexts from both sides. BiLSTM models perform better than BoE models since the former can model more context and preserve word order information.

To investigate the effect of pretrained word embeddings, we initialize the word embedding parameters with random initialization, GloVe pretrained word embeddings, and ELMo models. Since the official release of pretrained ELMo model has the output dimension of 1024, we do a linear

[4] http://pytorch.org

Model	dev F1	Model	dev F1
BoERand	54.60		
LSTMRand	50.51	BiLSTMRand	56.43
LSTMMean	50.15	BiLSTMMean	61.95
LSTMMax	51.49	BiLSTMMax	62.26
LSTMAttn	50.17	BiLSTMAttn	61.92

Table 6: Experiment results of Context Model using different pooling functions.

Model	dev F1	test F1
random	62.26	56.11
GloVe fixed	63.09	62.10
GloVe tuned	61.37	57.75
ELMo	**73.16**	**66.35**

Table 7: Experiment results of Context Model using different pretrained word vectors.

Model	KNOWN		ALL	
	dev F1	test F1	dev F1	test F1
random guess	38.81	38.04	37.39	32.96
rule-based	69.14	58.97	70.45	61.54
entity	73.97	**70.97**	61.07	60.06
context	74.29	63.11	73.16	66.35
combined	**81.75**	68.67	**76.13**	**69.92**

Table 8: Experiment results of baseline models and three proprosed models on KNOWN data and ALL data.

	CTX Wrong	CTX Correct
CMB Wrong	0/0	31/19 (II)
CMB Correct	98/127 (I)	413/406

Table 9: Model comparison between Context Model (CTX) and Combined Model (CMB) on dev/test sets.

transformation to reduce the output dimension to 100. The transformation matrix is part of the model parameters and are tuned during training. For GloVe, we use fixed and tuned versions of embeddings. For ELMo model, since it slows down the training significantly, we do not tune the model parameters. The results using BiLSTM+Max pooling are shown in Table 7. Fixed GloVe vectors improve the F1 score on dev and test set by 0.83% and 4.99%, respectively. The tuned version of GloVe does not improve that much, due to the fact that the dataset is small and too many parameters will cause overfitting. The pretrained ELMo model improves the F1 on the dev and test set by 10.90% and 10.24%, respectively, compared with random initialization, proving the powerfulness of ELMo model.

Results

Table 8 details the final result of the Entity Model, Context Model, and Combined Model. For comparison, we also include a baseline of random guessing and rule-based model. For the Context Model, we use the BiLSTM+Max pooling and use ELMo model to obtain words representations.

Our rule-based model outperforms random guess at a large margin. Surprisingly, it achieves a higher result than the Entity Model on ALL data. The reason is that the Entity Model cannot generalize to new entities, therefore performs badly on ALL data with many unknown entities. Compared with the Entity Model on the KNOWN data, Entity Model performs better.

For the **Entity Model**, the performance on KNOWN entities is better than that on ALL entities. This result illustrates that the model learns prior information about the entities in the train set. From a visualization of entity embeddings using tSNE (Maaten and Hinton 2008), we find that entities with similar political backgrounds tend to be close to each other in the embedding vector space. For example, Wall Street and McCain are close to each other, which is intuitive since they were both blamed by Obama, and they both blamed Obama and Bernanke, according to the training data. In addition, Obama, Congress, and Bernanke also form a cluster, since they were all blamed by McCain and blamed the Fed.

For the **Context Model**, we can see that it can effectively extract blame ties from news articles without prior entity knowledge. The F1 on the KNOWN test set is 63.11%; while on ALL test set, the F1 does not decrease as in the Entity Prior Model, it even increases to 66.35%. Unlike knowledge about entities, linguistic knowledge generalizes robustly to unseen test data, where most entities do not exist in the training data. This implies that our model can be used to extract blame ties in other occasions where the political settings are highly different. For example, when the President of the United States changes, we can still use our model to predict how the new President will play the blame game.

For the KNOWN entities, the **Combined Model** does *not* perform better than Entity Model. This shows that entities information alone may be useful in extracting blame ties than using contexts. However, such information could not be used when the entities are new, for instance, our ALL data. On the ALL data, the Combined Model achieves the best result on dev and test set, showing that the model can integrate context information as well as the entity prior information to make better predictions.

Therefore, the Context Model is the most robust one, applicable to extract blame tie among new entities. Entity prior information is helpful. If available, it can be used to boost the model performance.

Analysis

To verify our hypothesis about why the Combined Model is better than the Context Model, we take several examples on which the Combined Model succeeds while the Context Model fails and vice versa. To evaluate the practicality and generalization of the model, we use our trained model to extract blame ties from several recent news articles.

Articles
Trump Accuses **Russia** of Helping **North Korea** Evade Sanctions (1)
President **Donald Trump** accused **Russia** in unusually harsh terms of helping **North Korea** evade United Nations sanctions intended to press the country to give up its nuclear and ballistic missile programs. (2)
“ **Russia** is not helping us at all with **North Korea**,” **Trump** said in an interview with Reuters on Wednesday (3). “What China is helping us with, **Russia** is denting. In other words, **Russia** is making up for some of what China is doing.”
Trump has leaned on China to curb its support for **North Korean** leader Kim Jong Un’s regime, and in exchange has so far laid off the punishing trade measures he promised against the U.S.’s largest creditor during his campaign.
North Korea’s weapons programs are **Trump**’s most urgent foreign crisis (4). He has vowed not to allow the country to develop a missile capable of carrying a nuclear warhead to the U.S. mainland, threatening war to prevent it if necessary. But Kim has plunged ahead, and his government made rapid advances with both its missile and nuclear technology after **Trump** took office.
Trump’s criticism of **Russia** (5) is striking because members of Congress have said in the past that he was too reluctant to criticize **Russia**’s foreign policy and too eager to establish good relations with President Vladimir Putin.

Major Entities Mentioned
Trump (e_1), Russia (e_2), North Korea (e_3)

Table 10: An article from *Bloomberg* published on January 18, 2018. Top: paragraphs of the article containing blame patterns. The blame entities are in **bold face**. Bottom: major entities appearing in the article.

target **source**	e_1	e_2	e_3
e_1	-	0.82	0.91
e_2	0.02	-	0.40
e_3	0.00	0.01	-

Table 11: Prediction result of the article in Table 10 using the Context Model. The number represents the probability that the blame ties exists between the two corresponding entities.

Case Study

To analyze the fine-grained differences between behaviors of the Context Model and the Combined Model, we evaluate the two models on samples of the test data, and divide the samples into four classes, as shown in Table 9.

Class I samples are those on which the Context Model fails while the Combined Model works correctly. These samples usually involve entities that appear frequently in the train set. For instance, we want to figure out whether *Obama* blames *Republicans* based on the article titled *Obama Issues Sharp Call for Reforms on Wall Street* (*The New York Times*, April 2010). From sentences such as “*The president and his allies have eagerly portrayed Republicans as handmaidens of Wall Street... Obama avoided incendiary language attacking Republicans...*” we can see that the blame tie (Obama, republicans) holds. The Context Model predicts that the blame tie exists with a confidence score of 0.45, while the Combined Model gives a score of 0.97. The reason may be that the Combined Model learns prior information about Obama and Republicans: Obama blamed Republicans before in the training data, which makes sense since he is a Democrat.

Class II samples are those on which the Combined Model fails, while the Context Model works correctly. In the *USA Today* article *Obama tells Wall Street to join in* (April 2010), the context is *“...The president said the financial crisis, which has cost more than 8 million jobs so far, was ‘born of a failure of responsibility from Wall Street all the way to Washington.’...”*. The Context Model predicts that Washington does not blame Wall Street, at the confidence level of 0.62, which is true. However, the Combined Model mistakenly predicts a probability of 0.88 that the blame exists. This is because the Combined Model overly relies on entity information. Nevertheless, in Table 9 this class of samples is much fewer than class I samples.

Generalization to New Cases

To validate the generalization of our model, we conduct further analysis on news articles beyond the time frame of the financial crisis. Since most entities in this new dataset do not appear in our financial crisis dataset, we use the context model for generalizability test. In particular, we manually annotate 13 recent articles containing 14 blame ties from Google News, mostly in January 2018, and use our pretrained Context Model to extract blame ties from the articles. The F1 on individual blame ties is 72.00% on this new test data, and 8 out of 13 articles are labeled correctly. The result is consistent with the result of our financial crisis test set. This further demonstrates that the linguistic patterns for blame are generalizable to new scenarios. The articles on which the model fails mainly contain blame patterns that have not be seen in our financial crisis dataset.

Table 10 shows one of these new articles from Bloomberg Politics. In this news, United States Presidents accused Russia of helping North Korea evade United Nations sanctions. The blame tie is reflected in the sentences (1) (2) (3) and (5). From (4) we can infer that Trump has a negative attitude towards North Korea because of the nuclear weapons. The results of the Context Model are shown in Table 11. The model successfully identifies the blame ties from Trump to Russia and Trump to North Korea, with a high confidence score, while the probabilities for the other entity pairs are low.

Conclusion

We investigated blame analysis, for which previous research looked at the evolution of blame frames, without systematically connecting the frames to the actors who produce them. Experiments show that neural networks are effective for identifying blame ties from news articles. Our approach can enable researchers to quantify the importance of a frame and to understand how and why it became prevalent. It can also enable researchers to study actors' position alignments over time. To facilitate such research, we release our code and model for automatic blame tie extraction.

Acknowledgments

We thank Zhiyang Teng and Jie Yang for a lot of helpful discussions, Yan Zhang for helping with the inter-annotator agreement evaluation, Yu Yuan and Gokayaz Gulten and my other colleagues for helping with the proofreading. Yue Zhang is the corresponding author.

References

Alicke, M. D. 2000. Culpable control and the psychology of blame. *Psychological bulletin* 126(4):556.

Bail, C. A. 2016. Combining natural language processing and network analysis to examine how advocacy organizations stimulate conversation on social media. *Proceedings of the National Academy of Sciences* 113(42):11823–11828.

Bamman, D., and Smith, N. A. 2015. Open extraction of fine-grained political statements. In *EMNLP*, 76–85.

Gentzkow, M., and Shapiro, J. M. 2010. What drives media slant? evidence from us daily newspapers. *Econometrica* 78(1):35–71.

Groseclose, T., and Milyo, J. 2005. A measure of media bias. *The Quarterly Journal of Economics* 120(4):1191–1237.

Hendrickx, I.; Kim, S. N.; Kozareva, Z.; Nakov, P.; Ó Séaghdha, D.; Padó, S.; Pennacchiotti, M.; Romano, L.; and Szpakowicz, S. 2009. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, 94–99. Association for Computational Linguistics.

Hinton, G. E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. R. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Hobolt, S. B., and Tilley, J. 2014. *Blaming europe?: Responsibility without accountability in the european union*. Oxford University Press.

Hood, C. 2010. *The blame game: Spin, bureaucracy, and self-preservation in government*. Princeton University Press.

Kingma, D., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Li, J., and Hovy, E. 2014. Sentiment analysis on the people's daily. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 467–476. Doha, Qatar: Association for Computational Linguistics.

Luttrell, D.; Atkinson, T.; Rosenblum, H.; et al. 2013. Assessing the costs and consequences of the 2007–09 financial crisis and its aftermath. *Economic Letter* 8.

Maaten, L. v. d., and Hinton, G. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research* 9(Nov):2579–2605.

Malhotra, N., and Kuo, A. G. 2008. Attributing blame: The public's response to hurricane katrina. *The Journal of Politics* 70(1):120–135.

Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J.; Bethard, S. J.; and McClosky, D. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, 55–60.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781.

Miwa, M., and Bansal, M. 2016. End-to-end relation extraction using lstms on sequences and tree structures. *arXiv preprint arXiv:1601.00770*.

Mohammad, S.; Kiritchenko, S.; Sobhani, P.; Zhu, X.; and Cherry, C. 2016. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 31–41. San Diego, California: Association for Computational Linguistics.

Nguyen, T. H., and Grishman, R. 2015. Relation extraction: Perspective from convolutional neural networks. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, 39–48.

Nicol, O. 2016. No body to kick, no soul to damn: Responsibility and accountability for the financial crisis (2007–2010). *Journal of Business Ethics* 1–14.

O'Connor, B.; Balasubramanyan, R.; Routledge, B. R.; and Smith, N. A. 2010. From tweets to polls: Linking text sentiment to public opinion time series. *ICWSM* 11(122-129):1–2.

Olmeda, J. A. 2008. A reversal of fortune: blame games and framing contests after the 3/11 terrorist attacks in madrid. *by A. Boin, A. McConnell, P. Hart. Cambridge: Cambridge University Press*.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543.

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

Preoţiuc-Pietro, D.; Liu, Y.; Hopkins, D.; and Ungar, L. 2017. Beyond binary labels: political ideology prediction of twitter users. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, 729–740.

Rule, A.; Cointet, J.-P.; and Bearman, P. S. 2015. Lexical shifts, substantive changes, and continuity in state of the union discourse, 1790–2014. *Proceedings of the National Academy of Sciences* 112(35):10837–10844.

Shaver, K. G. 2012. *The attribution of blame: Causality, responsibility, and blameworthiness*. Springer Science & Business Media.

Tourish, D., and Hargie, O. 2012. Metaphors of failure and the failures of metaphor: A critical study of root metaphors used by bankers in explaining the banking crisis. *Organization Studies* 33(8):1045–1069.