Distinguishing Between Foreground and Background Events in News

Mohammad Aldawsari[a,b], **Adrian Perez**[b], **Deya Banisakher**[b], **& Mark A. Finlayson**[b]
[a]Department of Computer Science
Prince Sattam Bin Abdulaziz University, Al Kharj, Saudi Arabia
[b]School of Computing and Information Sciences
Florida International University, Miami, FL 33199, USA
mohammed.aldawsari@psau.edu.sa,
{apere946, dbani001, markaf}@fiu.edu

Abstract

Determining whether an event in a news article is a foreground or background event would be useful in many natural language processing tasks, for example, temporal relation extraction, summarization, or storyline generation. We introduce the task of distinguishing between foreground and background events in news articles as well as identifying the general temporal position of background events relative to the foreground period (past, present, future, and their combinations). We achieve good performance (0.73 F_1 for background vs. foreground and temporal position, and 0.79 F_1 for background vs. foreground only) on a dataset of news articles by leveraging discourse information in a featurized model. We release our implementation and annotated data for other researchers[1].

1 Introduction

Grimes et al. (1975) defined foreground events as the events that form the skeleton of a story whereas background events add supporting information. Ability to automatically extract such a distinction could guide document understanding and potentially be helpful in many natural language processing tasks such as temporal relation extraction (Naik et al., 2019) , summarization (Zhang et al., 2018), and storyline generation (Zhou et al., 2018). We introduce the task of distinguishing between foreground and background events, as well as identifying the general temporal position of backgrounds events relative to the foreground period (past, present, future, and their combinations). Identifying the general temporal position is a coarser analog to detailed, pairwise temporal relation extraction, and provides an intermediate step to ease the integration of discourse information into temporal understanding of the text.

Following (Grimes et al., 1975), we define *foreground* events as those that comprise the main topic of a news article, as indicated by the headline. In contrast, *background* events add supporting or contextual information. Note that while the document creation time (DCT) usually occurs after the foreground period, there is no reason why the DCT could not appear within or before it; our approaches does not assume any particular relationship between the DCT and the foreground period. Figure 1 shows a snippet of text where foreground events in red, and background events in other colors, divided into six general temporal position categories as illustrated in Figure 2 and defined in Table 1.

Background Past (BPast) events end before the foreground events begin.
Background Past Present (BPastPres) events start before and continues during the foreground period.
Background Present (BPres) events happen within the foreground event period.
Background Present Future (BPresFut) events begin during the foreground period and continue in the future.
Background Future (BFut) events begin after the foreground event period.
Background Past Present Future (BAll) events begin in the past, continue during the foreground period, and into the future.

Table 1: Background Event Categories, which are distinguished by their temporal position relative to the foreground period.

[1] The code and data are available at https://doi.org/10.34703/gzx1-9v95/NAMTMS

Proceedings of the 28th International Conference on Computational Linguistics, pages 5171–5180
Barcelona, Spain (Online), December 8-13, 2020

*A **car bomb**$_1$ **damaged**$_2$ half a city block in Istanbul Tuesday while the Prime Minister **attended**$_3$ a peace **conference**$_4$, which is scheduled from Monday to Wednesday. No **casualties**$_5$ were **reported**$_6$. The terrorist group behind the **attack**$_7$ has been **on the run**$_8$ from the military since the first major **bombing**$_9$ in 1998. The group **promised**$_{10}$ more **bombings**$_{11}$ soon, while the military **said**$_{12}$ that special security have been **implemented**$_{13}$ and would remain in place for the foreseeable future.*

Figure 1: An example text with foreground events marked in red, and background events in other colors, as defined in Figure 2.

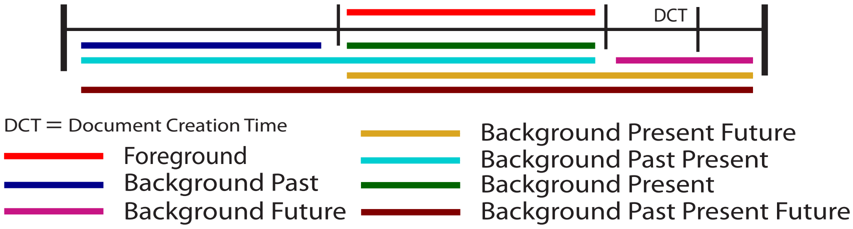

Figure 2: An illustration of the relative temporal position of foreground events in relation to background event categories. The document creation time (DCT) is here assumed to occur after the foreground events, but this is not strictly necessary.

The task is to classify an event as *Foreground*, *Background*, or *Other*, and additionally assign background events to one of the six possible general temporal positions relative to the foreground period. We assumes events are provided through some other process. The *Other* category includes events that are neither foreground nor background, such as generics or reporting events (e.g., *reported*$_6$ in Figure 1).

Our contributions are as follows: (1) we introduce a new task, namely, distinguishing foreground and background events and marking the general temporal position of background events relative to the foreground period; (2) we provide an annotated corpus with high inter-annotated agreement; (3) we demonstrate a simple featurized model that achieves reasonable performance (0.73 F_1 for background vs. foreground and temporal position, and 0.79 F_1 for background vs. foreground only); and (4) we show the utility of this task for three different NLP tasks—subevent detection, event coreference resolution and temporal relation extraction—by showing improvements in performance between 1 and 5 points of F_1.

The paper proceeds as follows. First we consider the prior work (§2), then describe the corpus and its annotations (§3). We next explain our model (§4) and its performance (§5), followed by a discussion (§6) that followed by an error analysis (§7). We show the utility of this information for downstream tasks (§8) and conclude by reiterating the contributions (§9).

2 Prior Work

Both Upadhyay et al. (2016) and Choubey et al. (2018) demonstrated approaches for identifying the central event in news articles. Upadhyay et al. (2016) proposed a rule-based system to identify the central event in a human-generated document summary. They evaluated their system on a human generated summaries from the New York Times Corpus (Sandhaus, 2008) where the central event had been identified. Similarly, Choubey et al. (2018) used several rule-based systems and statistical classifiers to identify the most important event in a news article. They trained and evaluated their systems on 30 news articles from the RED corpus (Mitamura et al., 2015) and 74 news articles from the KBP 2015 corpus (O'Gorman et al., 2016). Both were focused only on identifying a single central event, whereas we seek to label all events in a document as either *Foreground*, *Background*, or *Other*.

Huang et al. (2016) demonstrated an approach to placing events in news articles into three coarse temporal categories: *Past* events that have already occurred; *On-Going* events that are currently happening; and *Future* events that may happen. In that work, the temporal category was relative to the document creation time (DCT) and did not distinguish between foreground and background events. In contrast, our work seeks to mark the general temporal position of all background events relative to the foreground period.

3 Corpus

We annotated 99 news articles from the Intelligence Community (IC) corpus (Hovy et al., 2013). The IC corpus contains 100 news article but one article was merely a list of events rather than being a narrative. We used the gold event mentions that had been annotated on the corpus. The definition of *event* in Hovy et al. follows that of TimeML (Pustejovsky et al., 2003; Sauri et al., 2006), which has been well studied and shown to be reliably annotatable:

> We mean both events and states when we say 'event'. A *state* refers to a fixed, or regularly changing, configuration of entities in the world, such as 'it is hot' or 'he is running'. An *event* occurs when there is a change of state in the world, such as 'he stops running' or 'the plane took off'. (Hovy et al., 2013, p. 21)

The first and second authors labeled each event in the IC corpus with one of eight categories: *Foreground*, *Other*, or six varieties of *Background* (listed in Table 1). Disagreement was adjudicated by the third author. Overall agreement was 0.69 Cohen's κ. Table 2 shows agreements for individual classes as well as the statistics of the corpus. Note that in the corpus *BAll* only occurred 5 times, and *BPresFut* not at all. Table 2 shows the characteristics of the corpus and label counts.

Articles	99	
Sentences	1,955	
Tokens	48,737	
Event Mentions	4,086	
Avg. Sentences / article	19.7	
Avg. Tokens / article	487.4	
Avg. Events / article	30	κ
Foreground	1,501	0.66
Background Past (BPast)	851	0.66
Background Past-Present (BPastPres)	365	0.61
Background Present (BPres)	89	0.21
Background Present-Future (BPresFut)	0	-
Background Future (BFut)	160	0.43
Background Past-Present-Future (BAll)	5	0.66
Other	1,115	0.90
Overall Markings / Agreement	4,086	0.69

Table 2: Corpus Statistics

4 Model

Our model is a straightforward featurized logistic regression classifier. The features can be divided into five categories: Lexical, Syntactic, Semantic, Discourse and Time.

Lexical and Syntactic Temporal signals (e.g., *after* or *before*) often occur before background events. We used the temporal signals list collected by Derczynski and Gaizauskas (2010). This feature is encoded as bag of signals capturing whether a temporal signal is present in the text between the target event and the immediate preceding event. For syntactic features, we use the major part of speech (POS), tense, and aspect, all encoded as one-hot vector. We used `spaCy` (Honnibal and Montani, 2017) to compute both lexical and syntactic features.

Semantic To capture semantics we computed an event contextualized representation using (Akbik et al., 2018)'s implementation of BERT model *bert-base-uncased* (Devlin et al., 2019). The vector for an event is defined as the weighted sum of all subword embeddings extracted from BERT's last layer. We also captured the semantic frame of the event using SEMAFOR (Das et al., 2010), encoded as one-hot vector.

	Fine-grained				Coarse-grained			
Model	**Class**	**Prec.**	**Rec.**	F_1	**Class**	**Prec.**	**Rec.**	F_1
Our Model	Foreground	0.75	0.71	0.73	Foreground	0.73	0.74	0.74
	BPast	0.67	0.65	0.66	Background	0.73	0.73	0.73
	BPastPres	0.52	0.60	0.56				
	BPres	0.18	0.17	0.18				
	BFuture	0.38	0.51	0.43				
	Other	0.94	0.92	0.93	Other	0.93	0.92	0.93
	macro$_{avg}$	0.57	0.59	0.58	**macro**$_{avg}$	0.80	0.80	0.80
	micro$_{avg}$	0.73	0.73	**0.73**	**micro**$_{avg}$	0.79	0.79	**0.79**
Baseline (MFC)	**macro**$_{avg}$	0.17	0.06	0.09	**macro**$_{avg}$	0.33	0.12	0.18
	micro$_{avg}$	0.37	0.37	0.37	**micro**$_{avg}$	0.37	0.37	0.37
Baseline (Coref)	**macro**$_{avg}$	0.21	0.14	0.15	**macro**$_{avg}$	0.42	0.34	0.35
	micro$_{avg}$	0.34	0.34	0.34	**micro**$_{avg}$	0.46	0.46	0.46

Table 3: Our model's performance on all classes. *Background* is abbreviated as (B).

Discourse We employed two discourse features: RST discourse relation and the position of the event's sentence in the text. Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) is useful for many NLP tasks including sentiment analysis (Bhatia et al., 2015), information extraction (Maslennikov and Chua, 2007), and subevent detection (Aldawsari and Finlayson, 2019). We used Feng-Hirst discourse parser (Feng and Hirst, 2014) to build a discourse tree of each text, and post-processed the output to build a graph (Neumann, 2015). For events after the first, we extracted the rhetorical relation between the target event mention and the immediately preceding event. This feature is encoded as a one-hot vector covering all 16 main relation classes. We also captured the position of the event's sentence in the discourse. This was encoded as real number, normalized to a value between 0 and 1 by the number of sentences in the article.

Time We compute the difference, in days, between the date of the event mention and the date in the first sentence. If there is no date in the first sentence, we use the document creation time. The date of the event mention is taken to be any date used as an argument to the event, or otherwise the nearest date that appears in the sentence; if the event has neither, we assume the difference is zero. We normalized both dates to a calendar value using HeidelTime utility (Strötgen and Gertz, 2013). The difference is then encoded as one-hot vector feature with three possible values: negative, zero, or positive.

Classifier We used a logistic regression classifier from `scikit-learn` package (Pedregosa et al., 2011) for classification over the gold annotated event mentions. The classifier handles multi-class classification using a one-vs-rest scheme. Most of the parameters were left at their default settings [2]. We addressed data imbalance (seen in Table 2), by using the `class_weight=balanced` parameter to assign a higher mis-classification penalty to the minority class. We conducted 5-fold cross-validation for the experiment.

5 Results

We evaluated model performance using both macro and micro F_1. We conducted two experiments: in the first (the *fine-grained* condition) we use all classes from Table 2 except for two (*BAll* and *BPresFut*[3]). In the second experiment (the *coarse-grained* condition) we collapsed all background classes into one. Table 3 shows our model's performance under both conditions. In addition to a most frequent class (MFC) baseline, we designed a strong baseline inspired by the observation that the central event of a document usually has many co-referential event mentions (Choubey et al., 2018). This baseline operates as follows: (1) Mark an event as *Foreground* if it is part of an event coreference chain and the length

[2] `penalty=l2,C=0.1,random_state=42, max_iter=1000 class_weight=balanced, solver=liblinear, multi_class=ovr.`

[3] We merged the *BAll* class with the *BPres* class due to the small number of examples, and *BPresFut* had no examples in the corpus.

	Foreground	BPast	BPastPresent	BPresent	BFuture	Other
Foreground	-	22%	44%	5%	20%	9%
BPast	59%	-	22%	9%	4%	6%
BPastPresent	48%	32%	-	4%	12%	4%
BPresent	60%	21%	9%	-	4%	6%
BFuture	53%	15%	22%	6%	-	4%
Other	31%	24%	18%	2%	25%	-

Table 4: Fine-grained labeling error percentage between actual labels (rows) and predicted labels (columns).

of that chain is longer than or equal to the average of the lengths of event coreference chains for each article (the event coreference chains are identified based on the IC gold annotation); (2) Mark an event as *Other* if it is a reporting event corresponding to the IC gold annotation; (3) Otherwise mark the event as *BPast* for the fine-grained condition (the most frequent Background class), or *Background* for the coarse-grained condition.

6 Discussion

As shown in Table 3 the model performance in the fine-grained condition is lower compared to the coarse-grained performance, which is not surprising given the increased number of classes (and thus reduced data) and general difficulty of detecting temporal relationships.

We investigated the importance of each of the four feature sets to our model under the fine-grained condition by retraining while leaving out one set at time. In order of importance, they are: semantic (35% performance loss), discourse (4%), time (2%), syntactic and lexical (2%). Apparently, the most important feature set is the semantic features. The BERT vector is the most important feature for all classes, but the frame feature contributed more to the *Other* class. This is because of most of the events in this class are reporting events and were captured by the *Statement* frame. In the discourse set, the event's sentence position and discourse relation contributed equally to the model. The time feature contributed most to the *BPast* class because the *BPast* events mostly associated with past temporal dates. The syntactic and lexical features were the least contributing features to the model. On the other hand, when we dropped the contextualized embedding, the syntactic features contributed the most. By replacing BERT embeddings with ELMo (Peters et al., 2018) and Fasttext (Bojanowski et al., 2017) embeddings under the fine-grained condition, the performance decreases by 4% and 13%, respectively. Therefore, because it is known that BERT and ELMo capture more syntactic information than Fasttext, we hypothesize that the syntactic features were mostly—but not completely—captured by the BERT contextualized embeddings.

Finally, The lowest performing class is the *BPresent* class, which is to be expected because of the low number of examples.

7 Error Analysis

Upon detailed inspection we were able to discern several error classes aside from the usual noise introduced by the various sub-components. We observe that the model wrongly classifies Foreground events as Background if the event appears towards the end of the article. In our analysis, we also observe that this mislabeling occurs when the event is referred to in conjunction with some sort of temporal reference. For example, in an article regarding the capture of two people, in the sentence, *"The captured bomb-maker, Sami Muhammad Ali Said al-Jaaf, was* **seized** *in Baghdad on Jan. 15"*, the word *seized* is labeled as a Background event even though it is directly tied to the foreground. As shown in Table 4, this mislabeling constitutes 91% of the foreground event labeling error. Similarly, Background events that appear early in the article are often mistaken for Foreground events. Our model mistaking Background events as Foreground comprises 91% of the model's background labeling error. The model also wrongly classifies foreground events as *Other* (9% of the foreground labeling error) if the event mention looks like a reporting event due to the missing sense (e.g., *claimed* is used in the construction *claimed lives*, but can be mistaken for a reporting event). Another common error was the lack of explicit discourse or temporal information (e.g., a date) for identifying background events.

Within the fine-grained labeling of background events, we see that errors occur mainly between the distinction between events that are *BPast* and *BPastPresent*. Of the fine-grained error, the mislabeling of *BPast* as *BPastPresent* was 22% of the error (see Table 4); the labeling of *BPastPresent* as *BPast* constituted 32%. This being the largest error in the sub-classification task makes sense given that the two classes are quite similar. Consider the the text shown in Figure 3. In this example, the *operations* event is a *BPast* event incorrectly labeled as *BPastPresent*. We believe this is due in part to model not being sensitive to the precursory descriptors like the word *suspended*. For without that descriptor it would imply that the *operations* event is still ongoing.

> *Israeli security forces have been on high alert to guard against possible terror attacks by Hamas, which has suspended* **operations** *against Israel since its spiritual leader and founder Sheikh Ahmed Yassin was released from Israeli jail last October.*

Figure 3: Example text showing an event which is subject to the common mislabeling of BPastPresent for BPast.

With regard to general temporal position, changes of tense related to the document creation time (i.e., an event is in the past relative to the DCT, but in the future relative to the foreground period), caused difficulties in distinguishing between *BFut* and *BPast*. Though this error did not occur frequently, the failure to distinguish between the two classes comprises 19% of the overall fine-grained Background error, as shown in Table 4. We see the model struggle with examples such as *"Another cell was uncovered* ***last fall*** *, when the police carried out an operation against a group of Algerian and Moroccan radicals who were believed to be* ***planning*** *an attack on Madrid 's High Court and perhaps other targets"*. Difficulties in distinguishing between background classes in general were often the result of the writer assuming some commonsense or world knowledge on the part of the reader to infer the temporal relationship.

8 Applications

To validate the importance of capturing background and foreground events as well as the temporal position of background events to the foreground events, we experimented with incorporating this feature into three different NLP tasks, namely: subevent detection, event coreference resolution and temporal relation extraction. The goal of this experiment was to measure the performance with and without including Foreground/Background fine-grained classes as features. Even though some of the experiments we developed along the way outperform the state of the art, the emphasis here is on the contribution of these features in these tasks. All experiments were performed under the fine-grained condition and all experiments' implementation are released.

Model	**Prec.**	**Rec.**	**F$_1$**
Aldawsari and Finlayson (2019)	0.45	0.56	0.50
+Fine-grained Labels	0.50	0.61	**0.55**

Table 5: Subevent experiment result.

Model	**Prec.**	**Rec.**	**F$_1$**
Liu et al. (2014)	0.48	0.59	0.53
Our System	0.52	0.84	0.65
+Fine-grained Labels	0.55	0.85	**0.67**

Table 6: Event coreference experiment result.

> *In another* **fatality**$_1$*, a Spanish military adviser, Gonzalo Perez Garcia, who* **fell**$_2$ *into a coma after being seriously* **wounded**$_3$ *in a shootout last month died Wednesday, the Spanish Defense Ministry said.*

Figure 4: Example of a text where the relationship between one event and two others events is misclassified without the fine-grained Foreground/Background feature.

Subevent Detection Task The subevent detection is the process of identifying when one event is a subevent of another. The subevent relationship is defined in terms of (e_i,e_j), where e_i and e_j are events: event e_j is a subevent of event e_i if e_j is spatiotemporally contained by e_i (Hovy et al., 2013). The task is to classify a pair of events into one of the three classes: *parent-child, child-parent or no relation (NoRel)*, corresponding to the direction in the discourse flow. For subevent and event coreference experiments, we used the IC coprus, annotated with both event coreference and subevent relation, and compare our experiment to the state of the art models reported on the corpus (i.e. we used the gold annotation of background and foreground events). In the subevent experiment, we completely re-implemented the state of the art in this task (Aldawsari and Finlayson, 2019) and used the same evaluation metric, we refer interested readers to the paper for more details. For each event of a pair of event, we include the Foreground/Background fine-grained features as one-hot vector. As shown in table 5, after including these features, the performance increases by 5%. We performed an analysis of the increase and found that these features helped the model to distinguish between pairs with complex structure. For example, in Figure 4, both **fell**$_2$ and **wounded**$_3$ were previously classified as subevents of **fatality**$_1$, but after including the fine-grained labels, the model learned the NoRel class between $_{e1}$ and ($_{e2}$, $_{e3}$) since the latter events are *BPast* events.

Event Coreference Task Event coreference is the task of determining whether two events refer to the same event in the real world. For this experiment, we adapted some of the features mentioned in (§4) for event pairs coreference classification. That is, we used the major POS, tense, aspect, semantic frames, discourse relation between pairs and semantic similarity between pair's BERT embedding. Also we include an argument feature to determine whether the pair's arguments corefer. We extracted and resolved the arguments of events using AllenNLP's semantic role labeling (Gardner et al., 2018; He et al., 2017) and coreference resolution (Lee et al., 2017). We train a pairwise logistic regression classifier from `scikit-learn` over the features using parameters shown in table 8. As mentioned earlier, we used the IC corpus and conducted 5-fold cross-validation for the experiment. As shown in Table 6, the pairwise performance increases by 2% after including the Foreground/Background fine-grained features. The first column in the table shows Liu et al.'s (2014) pairwise model, the state of the art model on IC corpus[4] trained on 65 documents of the IC corpus which we use as a baseline. A shallow error analysis after including Foreground/Background fine-grained features reveals that almost all corrected cases were false negative and the events involved are foreground events. This observation indicates that detecting foreground events could be a useful intermediate step for event coreference improvement.

	TDD-Auto			TDD-Man		
Model	Pre.	Rec.	F1	Pre.	Rec.	F1
MAJOR	0.34	0.32	0.33	0.37	0.36	0.37
CAEVO	0.61	0.32	0.42	0.32	0.10	0.16
BiLSTM	0.55	0.48	0.52	0.24	0.23	0.24
Ning et al. (2017)	0.46	0.45	0.46	0.23	0.23	0.24
Our System	0.60	0.60	0.60	0.42	0.42	0.42
+Fine-grained Labels	0.61	0.61	**0.61**	0.42	0.42	**0.43**

Table 7: The first four models are an adaptation of state-of-the-art temporal models on TDD-Auto and TDD-Man reported by (Naik et al., 2019). The last two rows show our model without and with Foreground/Background fine-grained features, respectively.

	hyper-parameters		
Task	multi_class	solver	C
Subevent	ovr	liblinear	0.01
Event coref.	multinomial	lbfgs	0.1
Temporal	ovr	liblinear	0.0001

Table 8: The hyper-parameters used in all experiments corresponding to the `scikit-learn`'s implementation of logistic regression.

Temporal Relation Extraction Task Last but not least, extracting temporal information from text is a challenging but important task in NLP. In this experiment, we target the extraction of temporal relation between events which is one of the fundamental tasks in temporal processing as identified in the series

[4] Note that Liu et al.'s (2014) is not the state of the art result in event coreference generally, but merely the best performing result on the IC corpus.

TempEval (TE) workshops (Verhagen et al., 2007; Verhagen et al., 2010; UzZaman et al., 2013). We used the recently published dataset TDDiscourse (Naik et al., 2019), an augmented dataset of TimeBank-Dense (Cassidy et al., 2014) focused on discourse-level temporal ordering and used the same set of temporal relations as TimeBank-Dense (i.e., *after*, *before*, *simultaneous* , *includes* and *is-included*). The annotation of the corpus consists of two sets: Manual annotation (TDD-Man) and Automatic inference (TDD-Auto), we experiment on both.

For this task, we designed a simple and effective approach by concatenating pair's BERT embedding[5], POS, tense and aspect as one vector. We trained a logistic regression classifier over these features using hyper-paramters shown in table 8. We followed Naik et al.'s (2019) split setup of train, validation and test sets and compared the performance of our model to all models reported on the corpus. Similar to our previous experiments, we add Foreground/Background fine-grained features to the model and measure the performance with and without these features. As shown in Table 7, our approach in general outperforms all models on both TDD-auto and TDD-man by 9% and 5%, respectively. The reason behind the low performance of the other models has been addressed in Naik et al. (2019), and is out of scope for this paper. With regard to our model, as shown in the table, adding Foreground/Background fine-grained features did not help much in improving the model performance. This is in fact expected due to the fact that the fine-grained model was trained on a closed-domain (i.e., Intelligence Community (IC) news articles), which is a small fraction—55% are IC news articles and 40% of these are broadcast news—in the TDDiscourse corpus test set.

9 Contributions

We have presented a novel task: distinguishing between foreground and background events, as well marking the general temporal position of background events relative to the foreground period. We provided an annotated dataset, demonstrated a simple, featurized logistic regression model that performs well on this task which relies heavily on discourse understanding and show the utility of this task for three different NLP tasks. Our error analysis shows that while our model's performance is reasonable, there is still room for improvement by the introduction of commonsense or world knowledge to aid in reasoning.

10 Acknowledgements

This work was supported in part by a Saudi Arabian Cultural Mission Fellowship to Mr. Aldawsari, and also in part by NSF CAREER Award IIS-1749917.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, pages 1638–1649.

Mohammed Aldawsari and Mark Finlayson. 2019. Detecting subevents using discourse and narrative features. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pages 4780–4790, Florence, Italy.

Parminder Bhatia, Yangfeng Ji, and Jacob Eisenstein. 2015. Better document-level sentiment analysis from RST discourse parsing. In *Proceedings of the 20th Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 2212–2218, Lisbon, Portugal.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Taylor Cassidy, Bill McDowell, Nathanel Chambers, and Steven Bethard. 2014. An annotation framework for dense event ordering. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA.

Prafulla Kumar Choubey, Kaushik Raju, and Ruihong Huang. 2018. Identifying the most dominant event in a news article by mining event coreference relations. In *Proceedings of the 16th Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2018)*, pages 340–345, New Orleans, LA.

[5] Event embedding is extracted the same way discussed in §4, Semantics

Dipanjan Das, Nathan Schneider, Desai Chen, and Noah A. Smith. 2010. Probabilistic frame-semantic parsing. In *Proceedings of Human Language Technologies: The 8th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2010)*, pages 948–956, Los Angeles, CA.

Leon Derczynski and Robert Gaizauskas. 2010. USFD2: Annotating temporal expresions and tlinks for tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval'10)*, pages 337–340, Los Angeles, CA.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 14th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019)*, pages 4171–4186, Minneapolis, MN, June.

Vanessa Wei Feng and Graeme Hirst. 2014. A linear-time bottom-up discourse parser with constraints and post-editing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 511–521, Baltimore, Maryland, June. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia, July.

Joseph E Grimes, Roy E Grimes, and Joseph Evans Grimes. 1975. *The Thread of Discourse*. Walter de Gruyter, New York.

Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. 2017. Deep semantic role labeling: What works and what's next. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, pages 473–483, Vancouver, Canada, July.

Matthew Honnibal and Ines Montani. 2017. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. `https://github.com/explosion/spaCy`; Last accessed on Nov 28 , 2019.

Eduard Hovy, Teruko Mitamura, Felisa Verdejo, Jun Araki, and Andrew Philpot. 2013. Events are not simple: Identity, non-identity, and quasi-identity. In *Proceedings of the Workshop on Events: Definition, Detection, Coreference, and Representation*, pages 21–28, Atlanta, GA.

Ruihong Huang, Ignacio Cases, Dan Jurafsky, Cleo Condoravdi, and Ellen Riloff. 2016. Distinguishing past, on-going, and future events: The eventstatus corpus. In *Proceedings of the 21st Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 44–54, Austin, TX.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *EMNLP*.

Zhengzhong Liu, Jun Araki, Eduard Hovy, and Teruko Mitamura. 2014. Supervised within-document event coreference using information propagation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4539–4544, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Mstislav Maslennikov and Tat-Seng Chua. 2007. A multi-resolution framework for information extraction from free text. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007)*, pages 592–599, Prague, Czech Republic.

Teruko Mitamura, Zhengzhong Liu, and Eduard H Hovy. 2015. Overview of TAC KBP 2015 event nugget track. In *Proceedings of the 8th Text Analysis Conference (TAC 2015)*, Gaithersburg, MD.

Aakanksha Naik, Luke Breitfeller, and Carolyn Rose. 2019. Tddiscourse: A dataset for discourse-level temporal ordering of events. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 239–249.

Arne Neumann. 2015. discoursegraphs: A graph-based merging tool and converter for multilayer annotated corpora. In *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, pages 309–312, Vilnius, Lithuania.

Qiang Ning, Zhili Feng, and Dan Roth. 2017. A structured learning approach to temporal relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1027–1037, Copenhagen, Denmark, September. Association for Computational Linguistics.

Tim O'Gorman, Kristin Wright-Bettner, and Martha Palmer. 2016. Richer event description: Integrating event coreference with temporal, causal and bridging annotation. In *Proceedings of the 2nd Workshop on Computing News Storylines (CNS 2016)*, pages 47–56, Austin, TX.

Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June. Association for Computational Linguistics.

James Pustejovsky, José Castaño, Robert Ingria, Roser Saurí, Robert Gaizauskas, Andrea Setzer, and Graham Katz. 2003. TimeML: robust specification of event and temporal expressions in text. In *Fifth International Workshop on Computational Semantics (IWCS-5)*, pages 1–11.

Evan Sandhaus. 2008. *The New York Times Annotated corpus*. Linguistic Data Consortium, Philadelphia, PA. LDC Catalog Number LDC2008T19.

Roser Sauri, Jessica Littman, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky. 2006. TimeML annotation guidelines, version 1.2.1. https://catalog.ldc.upenn.edu/docs/LDC2006T08/timeml_annguide_1.2.1.pdf.

Jannik Strötgen and Michael Gertz. 2013. Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2):269–298.

Shyam Upadhyay, Christos Christodoulopoulos, and Dan Roth. 2016. "Making the News": Identifying noteworthy events in news articles. In *Proceedings of the 4th Workshop on Events*, pages 1–7, San Deigo, CA.

Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. 2013. Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9.

Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. 2007. Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the fourth international workshop on semantic evaluations (SemEval-2007)*, pages 75–80.

Marc Verhagen, Roser Sauri, Tommaso Caselli, and James Pustejovsky. 2010. Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 57–62.

Xingxing Zhang, Mirella Lapata, Furu Wei, and Ming Zhou. 2018. Neural latent extractive document summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 779–784, Brussels, Belgium, October-November. Association for Computational Linguistics.

Deyu Zhou, Linsen Guo, and Yulan He. 2018. Neural storyline extraction model for storyline generation from news articles. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1727–1736, New Orleans, Louisiana, June. Association for Computational Linguistics.