Event Prominence Extraction Combining a Knowledge-Based Syntactic Parser and a BERT Classifier for Dutch

Thierry Desot, Orphée De Clercq, and Veronique Hoste
LT[3], Language and Translation Technology Team
Ghent University, Groot-Brittanniëlaan 45, 9000 Gent, Belgium
{thierry.desot,orphee.declercq,veronique.hoste}@ugent.be

Abstract

A core task in information extraction is event detection that identifies event triggers in sentences that are typically classified into event types. In this study an event is considered as the unit to measure diversity and similarity in news articles in the framework of a news recommendation system. Current typology-based event detection approaches fail to handle the variety of events expressed in real-world situations. To overcome this, we aim to perform event salience classification and explore whether a transformer model is capable of classifying new information into less and more general *prominence* classes. After comparing a Support Vector Machine (SVM) baseline and our transformer-based classifier performances on several event span formats, we conceived *multi-word* event spans as *syntactic* clauses. Those are fed into our prominence classifier which is fine-tuned on pre-trained Dutch BERT word embeddings. On top of that we outperform a pipeline of a Conditional Random Field (CRF) approach to event-trigger word detection and the BERT-based classifier. To the best of our knowledge we present the first event extraction approach that combines an expert-based syntactic parser with a transformer-based classifier for Dutch.

1 Introduction

Recently, news publishers have shifted from newspapers to digital means which provide news readers easy access to a wide range of news information. However, the challenge is to find the right content that also corresponds to the user's personal interests. Therefore, many of today's major media and news websites offer automated news *recommendation* and *personalization* (Das et al., 2007; Odić et al., 2013; Moreira et al., 2019; Feng et al., 2020). News personalization paradigms define good news recommendations in terms of *similarity* to the user's previous reading behaviour. Hence, news articles are recommended based on *proximity* to other articles the user has read (Liu et al., 2010; Adnan et al., 2014). However, this contrasts with the normative concept of journalism that stimulates *diversity* of topics and events in unfiltered news streams (Pariser, 2011; Joris et al., 2019). In this study we consider the *news event* as a means to model both diversity and similarity in news articles in the context of a news recommendation system.

We present an event extraction approach that will be integrated in a news recommender for Dutch[1]. As current typology-based event detection fails to handle the variety of events in real-world situations we applied event *prominence* classification. This allows us to detect unrestricted news events and to overcome the sparsity of a small training data set. Our event extraction approach combines an expert-based syntactic parser with a transformer-based classifier:

- Input sentences are first pre-processed using a rule-based syntactic parser in order to generate smaller syntactic clauses as *multi-word* event spans.
- In a second phase, event *prominence* classification is applied in order to express whether it is a `main` or `background` event, using a classifier which is fine-tuned on pre-trained Dutch BERT word embeddings.

We also motivate the use of syntactic clauses as event spans, by comparing baseline and target classifier performances on other *multi-word* event span formats. On top of that we outperform a pipeline of a CRF event-trigger word detection approach

[1]https://www.ugent.be/mict/en/research/NewsDNA is an interdisciplinary research project at Ghent University that aims to outline a news recommendation algorithm driven by diversity of topics and events that occur in unfiltered news streams.

Proceedings of Recent Advances in Natural Language Processing, pages 346–357
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_040

and our BERT-based classifier. Furthermore, our approach is positioned with respect to the state of the art in Section 2 and is outlined in Section 3. An overview of the data set is given in Section 4. Section 5 presents the results of experiments on the held-out test set followed by a results analysis and discussion, conclusion and outlook on future work.

2 Related Work

Knowledge-based approaches are still frequently used for event extraction. Such methods are based on ontologies (Frasincar et al., 2009; Schouten et al., 2010; Arendarenko and Kakkonen, 2012) or rule-sets (Valenzuela-Escárcega et al., 2015) which represent expert knowledge. Information is mined from corpora based on lexical, syntactic (Hearst, 1992, 1998) and semantic patterns or frames (Cunningham, 2002a,b; Xie et al., 2013; Borsje et al., 2010; Hogenboom et al., 2013).

As the manual creation of rule-sets and ontologies is difficult and time-consuming, *data-driven* event extraction approaches made their entrance. The *ACE* (Automatic Context Extraction) annotation standards[2], *ERE* (Entities, Relations, Events) annotation standards (Song et al., 2015; Aguilar et al., 2014) and *TAC-KBP* (Text Analysis Conference Knowledge Base Population)[3] workshops and competitions stimulated the creation of data sets labeled with entities and events, e.g. the ACE 2005 corpus (Walker et al., 2006). As a consequence, supervised methods became predominant but initially concentrated on *fixed* event types using *single-word* event spans (Mitamura et al., 2015a). As compensation for small event spans, sentence or cross-sentential *context* information was used. In Ji and Grishman (2008) and Hong et al. (2011) events were extracted through cross-document and cross-sentence inference, respectively. Liao and Grishman (2011) improved event extraction performances adding topic classification information.

As *feature engineering* approaches emerged, a larger scope than one-word event spans was targeted. Hand-designed sets of lexical, semantic or syntactic features were extracted and fed into classifiers, allowing the model to take more context into account (Patwardhan and Riloff, 2009). Event extraction tasks are typically applied in a *pipeline* architecture where event trigger word identification, argument and event classification are conceived as separate tasks (Ahn, 2006). Other than a pipeline architecture, *multi-task architectures* perform several subtasks simultaneously to benefit from their interdependencies. In Li et al. (2013) events were extracted incorporating features that capture dependencies of multiple triggers and arguments. Luan et al. (2019) and Wadden et al. (2019) extracted events combined with named entity and argument role prediction.

However, the choice of features is a manual and elaborate process that requires extensive linguistic domain expertise. More recently *deep neural networks* superseded methods that show a strong dependency on feature resources, although the latter ones are still not definitely outperformed. Jacobs et al. (2018) and Nugent et al. (2017) used lexical, syntactic features, word2vec (Mikolov et al., 2013), glove (Pennington et al., 2014) and fastText (Bojanowski et al., 2017) word embeddings. Better performances were reported for an SVM classifier compared to a Recurrent Neural Network (RNN). In contrast, Nguyen and Grishman (2015) demonstrated that Convolutional Neural Networks (CNN) significantly outperformed feature-based methods on the ACE 2005 task.

Meanwhile, *contextual language models* have proven successful in a *transformer architecture* (Vaswani et al., 2017) that fully benefits from the attention mechanism. It has been integrated in a range of NLP tasks using pre-trained contextual *BERT* (Bidirectional Encoder Representations from Transformers) word embeddings (Devlin et al., 2018), predominantly for English. Mao and Liu (2019) report encouraging results for an event factuality classifier using BERT. Piskorski et al. (2020) report SVM event classifications with *Term Frequency-Inverse Document Frequency* (TF-IDF) that are outperformed by a fine-tuned BERT event classifier. The results of these studies inspired us to combine an expert-based syntactic parser with a BERT-based language model classifier for Dutch in order to extract *multi-word* events.

3 Method

3.1 Event Extraction for News Recommendation

In this study, an event is considered as the unit to measure *proximity* to other articles the user has read for news recommendation. It can be defined as the smallest extent of text that expresses its occurence

[2]https://www.ldc.upenn.edu/collaborations/past-projects/ace

[3]https://www.ldc.upenn.edu/collaborations/past-projects/tac-kbp

(Song et al., 2015), or a change of state at a particular place and time (Mitamura et al., 2015b), and is identified by a word or phrase called event *trigger, nugget, event span or mention*. Event mentions can be *single-word* event triggers that are usually (main) verbs, nouns, adjectives and adverbs. *Multi-word* event triggers can be *continuous* when the event span consists of consecutive tokens and even complete sentences, or *discontinuous* when its participants, or argument roles are also involved (Doddington et al., 2004). As they are more challenging to predict, we initially performed event classification on event spans with a *fixed and short* length, i.e. 5 token windows with a verbal head only. In a *second* phase we targetted longer events with a *variable* length, i.e. annotated events and *syntactic clauses* (Sections 5.1 and 5.2). The event extraction process in this study consists of automatically assigning an event *prominence* label to *continuous multi-word* event spans from a held-out test set. For the Dutch *input document* (translated in English) in Figure 1, the `Main` event is about a promotion campaign activity; the `Background` event provides background information about the `Main` event. Our hypothesis is that our target transformer classifier model is capable of categorizing new information into more general prominence classes, fine-tuned on pre-trained BERT word embeddings.

3.2 Syntactic Pre-Processing and Extraction of 5 Token Windows

Multi-word event spans, in this study defined as syntactic clauses as output from raw sentences processed by the *Alpino syntactic parser*, are fed into our baseline and target event prominence classifiers. The complete process is depicted in Figure 2.

The Alpino parser's knowledge-based part consists of a rule-based head driven phrase structure grammar (HPSG) and lexicon (100,000 entries). The integrated part-of-speech (POS) tagger reduces lexical ambiguity. The resulting dependency parse trees are disambiguated with a maximum entropy component (Van der Beek et al., 2002; Van Noord et al., 2006; Smessaert and Augustinus, 2010). An F-score of 91.14% was measured for 1,400 manually annotated sentences from the Twente News corpus (Ordelman et al., 2007).

For our experiments we applied a set of rules on the parser output in order to split sentences in the *test* set into separate *main and subclauses*. Subclauses in sentence medial position were not considered, but only in sentence initial and final position. In this way, the syntactic structure of our pre-processed test sentences is more similar to the clauses in the training set. For the Dutch sentence[4] in row 1 of Table 1, the labels `ssub` (*subclause*), *begin* and *end* position are used to extract the relative subclause from the syntactic parser output in row 3. As a preparatory step event classification was first performed on fixed event spans with a short length. To that end main *head verbs* in a *5 token window context* were extracted from the annotated events in our data, also by applying rules on the syntactic parser output.

We compared our syntax-driven event extraction approach with a **CRF**[5] (Lafferty et al., 2001) model to event detection as outlined in Colruyt et al. (under review), combined with our target classifier. For an input sequence of lexical, word shape and syntactic features, the CRF predicts a target sequence in *IOB* format. Tokens starting an event mention are labelled as *B*, tokens inside the mention as *I*, and tokens outside the mention are labeled as *O*.

Raw input sentence
`Soldaten zullen worden ingezet in de wijk Rocinha die zo'n 70.000 inwoners telt`
Begin and end position of a subclause
`<begin=9 cat=ssub end=13>`
Extracted (relative) subclause
`die zo'n 70.000 inwoners telt`

Table 1: Subclause extracted from syntactic parser output

3.3 Baseline Classification Models

For a prominence classification of multi-word event spans, i.e. 5 token windows or syntactic clauses, into `Main`, `Background` and `None` event labels, an **SVM** classifier was trained as baseline model using the `scikit-learn` Python library. SVM performances were compared for *Bag of Words* (BOW) and TF-IDF *count-based* methods. Instead of deriving meaning from an entire corpus, word representations are constructed one sentence at a time, with a *prediction-based* method that predicts word identity given a sentence context. The model

[4] English translation: *"Soldiers will be deployed in the Rocinha district, which includes about 70,000 inhabitants."*

[5] https://sklearn-crfsuite.readthedocs.io/en/latest/index.html

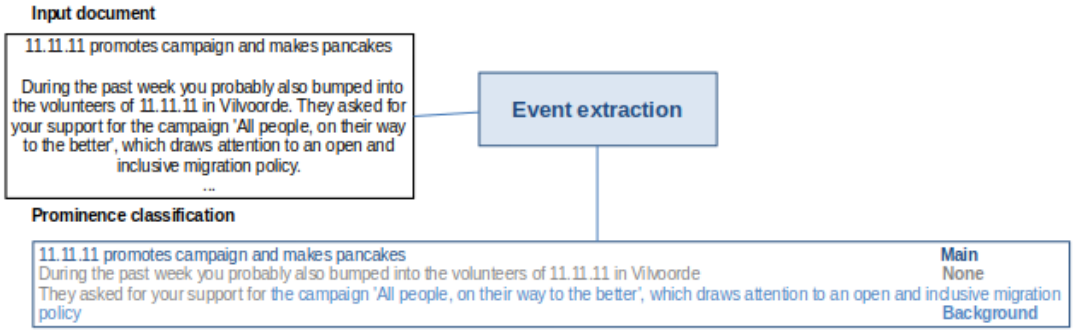

Figure 1: Example of event prominence classification in order to extract main or background events

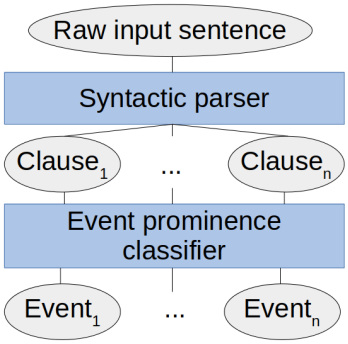

Figure 2: Raw sentences split into syntactic clauses are the input of the event prominence classifier

learns that words occurring in similar sentence contexts are semantically related. This was applied by combining the SVM classifier with Dutch pre-trained `word2vec` word embeddings (Tulkens et al., 2016). The embeddings were pre-trained on the combined Dutch Roularta[6], Wikipedia[7] and SoNaR corpora (Oostdijk et al., 2013) with a total of 54.8 million sentences and 803 million words.

3.4 Transformer-Based Target Classification Model

SVM baseline performances for event prominence classification were compared with a transformer-based (Section 2) classifier that relies entirely on the *self-attention* mechanism. It relates different positions of a single sequence in order to compute a representation of the sequence (Vaswani et al., 2017). For an input sequence $x = (x_1, ..., x_n)$ of n elements, where $x_i \in \mathbb{R}^{d_x}$ each attention head in the self-attention sublayers calculates a sequence $z = (z_1, ..., z_n)$, where $z_i \in \mathbb{R}^{d_z}$. Each output element, z_i, is computed as weighted sum of linearly *transformed* input elements,

$$z_i = \sum_{j=1}^{n} \alpha_{ij} \left(x_j W^V\right) \tag{1}$$

Each weight coefficient, α_{ij}, is calculated with a softmax function,

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{n} \exp(e_{ik})} \tag{2}$$

and e_{ij} is computed with a function comparing two input elements,

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K)^T}{\sqrt{d_z}} \tag{3}$$

where W^Q, W^K and $W^V \in \mathbb{R}^{d_x \times d_z}$ are parameter indices that are unique per layer and attention head. The attention function maps vectors of queries W^Q and key-value pairs W^K, W^V to an output (Shaw et al., 2018).

BERT are unsupervised deep bidirectional word embeddings (Devlin et al., 2018) pre-trained on large corpora in the target language. Frequently, a smaller dataset is used for fine-tuning for the target NLP task. A replication study and evaluation of BERT resulted in *RoBERTa* (Liu et al., 2019) that is trained on more data, bigger batches and longer sequences. Bidirectional pre-training is realized with a *masked language model* (MLM). The MLM randomly masks input tokens in order to predict the original vocabulary relying on its left and right context. In addition to the MLM *next sentence prediction* (NSP) jointly pre-trains text-pair representations.

A Dutch BERT model, **BERTje** (de Vries et al., 2019) has been pre-trained on a dataset of 2.4 billion tokens from Wikipedia, Twente News Corpus (Ordelman et al., 2007), and SoNaR-500 corpora (Oostdijk et al., 2013). *RobBERT* (Delobelle et al., 2020), a RoBERTa based and larger model has

[6] www.roularta.be/en
[7] wikimedia.org/nlwiki/20150703

Events	#	Entities	#	Item	#
Main	4248	PER	6943	Vocabulary	13276
Backgr.	3154	LOC	5537	Tokens	90062
None	1824	ORG	4441	Sentences	6924
		MISC	490	Documents	1771
Total	9226		17411		

Table 2: EventDNA corpus statistics

been pre-trained on 6.6 billion Dutch tokens from the OSCAR corpus (Suárez et al., 2019). Other than BERTje, RobBERT does not integrate NSP. Both models have an architecture of 12 transformer blocks (bidirectional layers) and 12 self-attention heads and a hidden size of 768.

4 Data

Our baseline and target event prominence classification models were trained on the EventDNA corpus. It comprises 1,771 Dutch news articles (Table 2, *Documents*), of which only the title and lead paragraph were kept, and is annotated with entities, news events and *IPTC* (International Press Telecommunications Council) Media Topic codes[8] (Colruyt et al., under review). The annotation protocol was based on the *ERE* (Entities, Relations, Events) annotation standards (Song et al., 2015; Aguilar et al., 2014).

Entity spans can be assigned one out of four possible labels: person (`PER`), location (`LOC`), organization (`ORG`), and (`MISC`) for other entity values (Table 2, *Entities*). A sentence can comprise more than one event (with an average of 1.3 events per sentence). All relevant semantic information (with priority over syntactic information) is included in the event span that can contain entire, main or subclauses, or nominal expressions. Hence the event's arguments can be included. An Event span is annotated with a *prominence* feature label: `Main` events bring new information and actually caused the reporter to write the article; `Background` events give context or background to the `Main` event; raw sentences without events are labeled as `None` events (Table 2, *Events*). Our motivation to apply prominence classification other than event *type* labeling is mainly driven by a prior analysis of the EventDNA corpus which revealed a high frequency (32%) of event types in a small data set (Table 2, *Sentences*) that cannot be classified into one of the event types specified in the EventDNA annotation protocol. Figure 1 presents an example[9] of an event span labeled as `Background` event, preceded by a `Main` and `None` event. For more information about the EventDNA annotations we refer the reader to Colruyt et al. (2019).

For our experiments, both data sets with *annotated events* and *5 token windows* with verbal head, extracted from the corpus, were randomized and split into 80% train, 10% development (DEV) and 10% held-out test data as shown in Table 3. The number of 5 token window instances in the training and test set is lower than the number of annotated events, as only events with a verbal head were extracted. Subsequently, performance comparisons between the models trained on those two data sets in Section 5.1 are not entirely fair. For that reason we provided a test set with only overlapping instances between the 5 token window instances and the annotated event instances for a fair comparison (Table 3, *Annotated events2*).

In order to verify the feasibility of our approach to classify events based on the test sentences, split into *syntactic clauses*, with the Alpino syntactic parser (Section 5.2), we counted the syntactic constituents in the training data annotated with events. Table 4 shows that the majority of the *annotated* events in the training set consist of a *single verbal* main-, subclause or infinitival construction. By splitting our test intput sentences into syntactic clauses the syntactic structure of our pre-processed test sentences is more similar to the *single verbal* main-, subclause or infinitival construction (50.97%) *and* main clauses combined with other verbal constituents (13.57%) in the training set.

As the test sentences were split into syntactic clauses, the number of test instances (*Syntactic clauses*) in Table 3 exceeds the number of the original *Raw* test sentences. Hence, performance comparisons on the *Raw sentences* and *Syntactic clauses* for the syntax based event extraction experiments in Section 5.2 are not entirely fair. However, the test sets in Table 3, used for our experiments in section 5, are based on the *same* 10% held-out test data from the EventDNA corpus. We mapped the *raw sentence* and *syntactic clause* test set versions with the *Annotated events* in order to assign the event labels, and manually verified these. For raw sentences comprising several events, we randomly assigned one event prominence class. We also pro-

[8] https://iptc.org/standards/media-topics/

[9] Dutch translation: *"Zij vroegen uw steun voor de campagne 'Allemaal mensen, onderweg naar beter', die aandacht vraagt voor een open en solidair migratiebeleid."*

Data set	Instances training set	Instances test set
Annotated *events*	7362	934
Annotated *events2*	7362	780
5 token *windows*	6248	780
Raw *sentences*	-	904
Syntactic *clauses*	-	1030
Syntactic *clauses2*	-	904

Table 3: Training and test sets - annotated events, 5 token windows, raw sentences and syntactic clauses

Single syntactic constituent	Annotated events Train set (%)
Non-verbal:	
Noun Phrase	35.44
Verbal:	
Infinitival construction	1.84
Main clause	44.93
Subclause	4.20
Main clause + verbal constit.	13.57

Table 4: Syntactic constituents in EventDNA training data

vided *Syntactic clauses2* for testing with the same number of instances as Raw *sentences*. In order to align both files we only kept one randomly selected syntactic clause per sentence in the former file.

5 Experiments and Results

We trained and tested our *SVM baseline* event classifier and *target BERT* event classifier on 5 token windows and annotated events (Section 4). Then we fed the syntactic clauses from the syntactic parser into the baseline SVM and target BERT classifiers. Finally, we positioned our approach w.r.t. a pipeline of a CRF approach to event-trigger word detection and target prominence classifier (Section 5.2).

5.1 Event Extraction Based on 5 Token Windows and Gold-Standard Events

For training the SVM baseline event classification models (Section 3.3), parameters were optimized using the DEV set. The best results were obtained with an RBF kernel with cost $C = 20$, using the default *scale* value of the parameter *gamma*, applying *one-vs-rest* classification. SVM performances are compared for BOW, TF-IDF, and pre-trained `word2vec` Dutch word embeddings. For fine-tuning the target BERTje and RobBERT prominence classifiers (Section 3.4), *AdamW* optimizer was used (Loshchilov and Hutter, 2017) with a learning rate of 1e-5 and a batch size of 10 instances. The maximum sequence length is similar to 69 tokens, which is the maximum sequence token length of the annotated events in the training data. As we are interested in *single sentence classification* we added the special [CLS] (classification) token. Minimal loss was obtained after 3 epochs of training for BERTje and 4 epochs for RobBERT with a cross entropy loss function. Performances were evaluated using Recall (*Rec.*), Precision (*Prec.*) and F-score.

Surprisingly, the SVM baseline classifier with `word2vec` embeddings did not outperform the SVM TF-IDF and BOW models (Table 5). However, the study of Tulkens et al. (2016) also reported varying performances for the Dutch `word2vec` embeddings compared to BOW and TF-IDF. In general, better performances are exhibited for the models trained on the annotated events than for the 5 token windows. For both data sets the transformer models outperform the SVM classifiers with slightly superior performances for RobBERT on the 5 token windows and BERTje on the annotated events. For the latter model, Table 6 exhibits worst performances on the `Background` prominence class, compared to `Main` and `None` classes.

5.2 Syntax Based Event Extraction

As we defined our *target* multi-word event spans as syntactic clauses (Section 3.1), the *raw* sentences in the test set were *pre-processed* with the syntactic parser outlined in Section 3.2, before feeding the resulting clauses to the baseline SVM and target BERTje classifiers as used in Section 5.1. Table 7 shows best performances for the BERTje classifier on *syntactic clauses*, that are very similar to *syntactic clauses 2*, the syntactic clauses that were aligned (Section 4) with the *Raw sentences* for a fair comparison.

We also compared our event extraction approach using the BERTje model that classifies multi-word event spans, conceived as syntactic clauses, with a pipeline consisting of a CRF for *event-trigger word detection* (Section 3.2) and our BERT-based classifier. The CRF model was trained for ten iterations on the annotated `Main` and `Background` events in the training set (Section 4) and tested on the raw sentences in the held-out test set (Table 3). Only

Model	5 token windows			Annotated events			Annotated events 2		
	Prec.	Rec.	F-score	Prec.	Rec.	F-score	Prec.	Rec.	F-score
SVM (BOW)	56.56	58.38	56.62	64.49	64.01	64.08	63.81	62.20	62.91
SVM (TF-IDF)	56.61	58.00	56.63	65.15	66.10	65.76	65.68	62.35	63.49
SVM (Word2vec)	52.96	53.64	53.24	60.13	59.23	59.92	60.42	57.87	58.93
BERTje	57.18	58.07	57.29	70.77	70.74	**70.75**	69.55	69.35	69.37
RobBERT	57.89	58.46	**58.13**	70.09	70.14	70.08	69.14	69.38	69.22

Table 5: SVM, BERTje and RobBERT event classification performances (%), trained and tested on 5 token windows and annotated events

Test set events (%)	Annotated events		
	Prec.	Rec.	F-score
Backg. (34.90)	68.01	69.11	68.32
Main (45.08)	71.24	70.45	70.56
None (20.02)	75.43	75.28	**75.33**

Table 6: BERTje classification performances (%) on annotated events per prominence class

the resulting detected (67% F1-score) `Main` and `Background` events, without the `None` events, were fed into the transformer classifier. Table 8 exhibits significantly poorer prominence classification results on the *CRF detected events* compared to classification on the syntactic clauses (also without the `None` events).

6 Results Analysis and Discussion

Analysis of BERTje attention heatmaps indicated the feasibility of our event extraction approach combining a syntactic parser and a BERT classifier. The sentence *"Then an adviser to the president was convicted because he lied"*[10] (Figure 3 - left) consists of a main clause *"Then an adviser to the president was convicted"*[11] (middle) with a `main` event, and a subclause *"because he lied"*[12], the `Background` event (right). Figure 3 (left) shows that most attention in the raw sentence is erroneously attributed to the past participle in the subclause, "gelogen" (lied). After splitting the sentence in its main and subclause, most attention is now correctly attributed to the verbs in the `Main` (middle) and `Background` (right) event. Although the BERTje classifier performances on the syntactic clauses are better, compared to the CRF detected events (Table 8), classification performances are still poorer compared to classification on the test set with annotated events (Table 5). As the training data has been annotated taking into account semantic information, with priority over syntactic information, the boundaries of the syntactic clauses generated by the Alpino parser, are frequently different from the boundaries of the annotated events which results in poorer performances. On top of that 35.44% of the EventDNA training data consists of non-verbal constituents (Table 4). These are mainly news article titles, but also noun phrases as part of a main clause that have been annotated as separate events. However, our rule-set on top of the syntactic parser, splits raw test sentences into separate *main and subclauses* (Section 3.2), but does not isolate nominal constituents. This also partially explains poorer performances on the syntactic clauses compared to the annotated test events. A possible solution for this bottleneck is combining the rule-set on top of the syntactic parser, with the BERTje self-attention mechanism. Tokens in the syntactic clause to which the highest attention values are attributed can be extracted, e.g. nominal constituents as part of a clause.

The transformer models outperform the SVM (Section 5) and benefit from the structure of language that is taught during pre-training. Certain self-attention heads exhibit linguistic notions of syntax and coreference. In line with the studies of Vig (2019), Vig et al. (2019) and Clark et al. (2019), coreference relations are situated in the middle and deeper layers of the self-attention blocks as depicted in Figure 4. For the sentence *"She survived the bullet to her head"*[13], coreference between the Dutch personal pronoun *ze* (she), on the right, and the possessive pronoun, on the left, *haar* (her) is depicted as connecting lines. Darker colors represent higher attention weights. In general

[10] Original Dutch sentence: *"Toen werd een adviseur van de president veroordeeld omdat hij gelogen had"*

[11] Original Dutch sentence: *"Toen werd een adviseur van de president veroordeeld"*

[12] Original Dutch sentence: *"omdat hij gelogen had"*

[13] Original Dutch sentence: *"Ze overleefde de kogel door haar hoofd"*

Model	Syntactic clauses			Syntactic clauses 2			Raw sentences		
	Prec.	Rec.	F-score	Prec.	Rec.	F-score	Prec.	Rec.	F-score
SVM (BOW)	52.62	53.21	52.26	50.42	52.85	51.49	48.30	52.45	50.10
SVM (TF-IDF)	52.12	54.81	53.64	53.06	55.52	54.00	52.66	55.08	53.66
SVM (Word2vec)	49.80	51.82	50.38	50.07	51.88	50.08	46.95	49.82	47.26
BERTje	62.65	62.95	**62.95**	59.01	62.16	60.73	53.24	57.19	**54.22**
RobBERT	58.42	60.17	59.10	57.43	60.09	58.30	51.04	53.87	52.61

Table 7: SVM, BERTje and RobBERT event classification performances (%), trained on annotated events, and tested on syntactic clauses and raw sentences

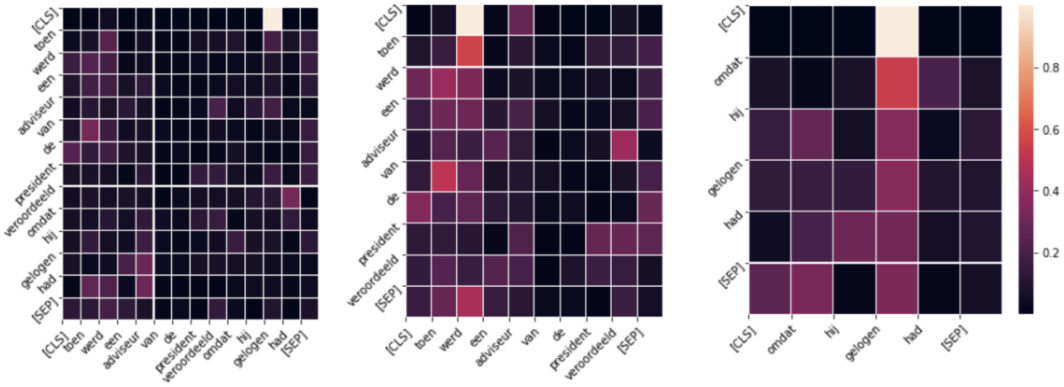

Figure 3: Heatmap with the highest attentions (lightest color) for the event verbs in the raw sentence (left), for the main clause (middle) and for the subclause (right)

BERTje	Prec.	Rec.	F-score
Syntactic clauses	66.48	60.71	**62.43**
CRF detected events	64.97	45.77	51.17

Table 8: BERTje classification (%) on CRF detected Main/Background events and syntactic clauses

Figure 4: BERTje - 12 x 12 self-attention blocks (right), coreference (left), layer 9 attention block 3

BERTje outperforms the baseline SVM classifier, however, the difference in performance is more outspoken on the annotated events (Table 5). This indicates that a transformer model benefits from processing coreference and long distance relationships in the *longer* annotated events.

In spite of the advantages of using the transformer model, minimal loss was already obtained after only 3 epochs of training for BERTje (Section 5.1). The BERTje model pre-trained on a large corpus, allows a small dataset being used for fine-tuning on the event prominence classification task. However, applying data augmentation on the small NewsDNA dataset might increase training time during fine-tuning. Although the pre-trained BERTje model is large (2.4 billion tokens), it contains other data than news corpora, whereas our training set consists entirely of news. This raises the question whether it is not better to use a domain-specific pre-trained model consisting entirely of news corpora.

A bottleneck of classifying prominence labels *only* based on the sentence level, is the lack of context information. This has an impact mainly on the `Background` prominence class (Table 6). Semantic and syntactic information cues within a sentence can in some cases be sufficient to correctly predict a *Background* class. E.g. the conjunction "when" in *"when she tried to convince the shooter"* introduces a subclause with a noun "shooter", which refers to a shooting or killing `Main` event *outside* the subclause that contains a `Background` event "convince". However, frequently more context information is necessary in order to correctly pre-

dict the `Background` prominence label. As a next research step, for fine-tuning the transformer model, extra separator tokens [SEP] with previous and/or next annotated events can be inserted to the current training instances. This can provide the model more context to improve `Background` prominence class predictions. Furthermore, instead of using event prominence classes, more generalized event types can be generated, by mapping the original more specific event types in the NewsDNA data to broader event classes. This would decrease the need for more context information. However, the latter approach might not offer the complete solution to handle the variety of events expressed in real-world situations.

7 Conclusion and Future Work

This study shows that an event extraction approach of an *expert-based* syntactic parser in combination with a *transformer-based* classifier (BERTje) is feasible. The resulting model outperforms (62.95% F-score) a pipeline of a CRF approach to event-trigger word detection and a BERT-based event classifier. We also demonstrated that a syntactic clause can be used as event span. Prominence classification is our answer to take into account a *real-world* situation where event types in held-out test data are frequently not covered because of training data scarcity. The BERTje model benefits from self-attention heads with linguistic notions such as syntax and coreference and outperformed (70.75% F-score) an SVM baseline classification model. A bottleneck of classifying prominence labels only based on the sentence level, is the lack of context. This has an impact mainly on the `Background` prominence class. Therefore further work includes exploring ways to provide more context information in the transformer model. It can be fine-tuned on training data where previous and following annotated events to the current single event instances are inserted. As a next step the BERTje self-attention mechanism will be leveraged to select the tokens in the syntactic clause with the highest attention values. This will allow e.g. the generation of nominal constituents on top of the clauses generated by the syntactic parser. Although the transformer model exhibits promising performances fine-tuned on a small dataset, data augmentation of the training set might optimize the fine-tuning and boost performances. Finally the classifier output will be fed into a news recommender system.

Acknowledgements

We thank the reviewers for their valuable comments. This work was supported by Ghent University under grant BOFGOA2018000601.

References

Md Nuruddin Monsur Adnan, Mohammed Rashid Chowdury, Iftifar Taz, Tauqir Ahmed, and Rashedur M Rahman. 2014. Content based news recommendation system based on fuzzy logic. In *2014 International Conference on Informatics, Electronics & Vision (ICIEV)*, pages 1–6. IEEE.

Jacqueline Aguilar, Charley Beller, Paul McNamee, Benjamin Van Durme, Stephanie Strassel, Zhiyi Song, and Joe Ellis. 2014. A comparison of the events and relations across ace, ere, tac-kbp, and framenet annotation standards. In *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 45–53.

David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1–8.

Ernest Arendarenko and Tuomo Kakkonen. 2012. Ontology-based information and event extraction for business intelligence. In *International Conference on Artificial Intelligence: Methodology, Systems, and Applications*, pages 89–102. Springer.

Leonoor Van der Beek, Gosse Bouma, Rob Malouf, and Gertjan Van Noord. 2002. The alpino dependency treebank. In *Computational linguistics in the netherlands 2001*, pages 8–22. Brill Rodopi.

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.

Jethro Borsje, Frederik Hogenboom, and Flavius Frasincar. 2010. Semi-automatic financial events discovery based on lexico-semantic patterns. *International Journal of Web Engineering and Technology*, 6(2):115–140.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*.

Camiel Colruyt, Orhpée De Clercq, and Veronique Hoste. under review. Eventdna: a dataset for dutch news event extraction as a basis for news diversification. Under review.

Camiel Colruyt, Orphée De Clercq, and Véronique Hoste. 2019. Eventdna: guidelines for entities and events in dutch news texts (v1. 0). *LT3 Technical Report-LT3 19-01*.

Hamish Cunningham. 2002a. Gate: A framework and graphical development environment for robust nlp tools and applications. In *Proc. 40th annual meeting of the association for computational linguistics (ACL 2002)*, pages 168–175.

Hamish Cunningham. 2002b. Gate, a general architecture for text engineering. *Computers and the Humanities*, 36(2):223–254.

Abhinandan S Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. 2007. Google news personalization: scalable online collaborative filtering. In *Proceedings of the 16th international conference on World Wide Web*, pages 271–280.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. Robbert: a dutch roberta-based language model. *arXiv preprint arXiv:2001.06286*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie M Strassel, and Ralph M Weischedel. 2004. The automatic content extraction (ace) program-tasks, data, and evaluation. In *Lrec*, volume 2, pages 837–840. Lisbon.

Chong Feng, Muzammil Khan, Arif Ur Rahman, and Arshad Ahmad. 2020. News recommendation systems-accomplishments, challenges & future directions. *IEEE Access*, 8:16702–16725.

Flavius Frasincar, Jethro Borsje, and Leonard Levering. 2009. A semantic web-based approach for building personalized news services. *International Journal of E-Business Research (IJEBR)*, 5(3):35–53.

M Hearst. 1998. Wordnet: An electronic lexical database and some of its applications. *Automated Discovery of WordNet Relations*.

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Coling 1992 volume 2: The 15th international conference on computational linguistics*.

Alexander Hogenboom, Frederik Hogenboom, Flavius Frasincar, Kim Schouten, and Otto Van Der Meer. 2013. Semantics-based information extraction for detecting economic events. *Multimedia Tools and Applications*, 64(1):27–52.

Yu Hong, Jianfeng Zhang, Bin Ma, Jianmin Yao, Guodong Zhou, and Qiaoming Zhu. 2011. Using cross-entity inference to improve event extraction. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 1127–1136.

Gilles Jacobs, Els Lefever, and Véronique Hoste. 2018. Economic event detection in company-specific news text. In *Proceedings of the First Workshop on Economics and Natural Language Processing*, pages 1–10.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *Proceedings of ACL-08: Hlt*, pages 254–262.

Glen Joris, Camiel Colruyt, Judith Vermeulen, Stefaan Vercoutere, Frederik De Grove, Kristin Van Damme, Orphée De Clercq, Cynthia Van Hee, Lieven De Marez, Veronique Hoste, et al. 2019. News diversity and recommendation systems: Setting the interdisciplinary scene. In *IFIP International Summer School on Privacy and Identity Management*, pages 90–105. Springer.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 73–82.

Shasha Liao and Ralph Grishman. 2011. Acquiring topic features to improve event extraction: in preselected and balanced collections. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 9–16.

Jiahui Liu, Peter Dolan, and Elin Rønby Pedersen. 2010. Personalized news recommendation based on click behavior. In *Proceedings of the 15th international conference on Intelligent user interfaces*, pages 31–40.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. *arXiv preprint arXiv:1904.03296*.

Jihang Mao and Wanli Liu. 2019. Factuality classification using the pre-trained language representation model bert. In *IberLEF@ SEPLN*, pages 126–131.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Teruko Mitamura, Zhengzhong Liu, and Eduard H Hovy. 2015a. Overview of tac kbp 2015 event nugget track. In *TAC*.

Teruko Mitamura, Yukari Yamakawa, Susan Holm, Zhiyi Song, Ann Bies, Seth Kulick, and Stephanie Strassel. 2015b. Event nugget annotation: Processes and issues. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 66–76.

Gabriel de Souza P Moreira, Dietmar Jannach, and Adilson Marques da Cunha. 2019. On the importance of news content representation in hybrid neural session-based recommender systems. *arXiv preprint arXiv:1907.07629*.

Thien Huu Nguyen and Ralph Grishman. 2015. Event detection and domain adaptation with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 365–371.

Tim Nugent, Fabio Petroni, Natraj Raman, Lucas Carstens, and Jochen L Leidner. 2017. A comparison of classification models for natural disaster and critical event detection from news. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 3750–3759. IEEE.

Ante Odić, Marko Tkalčič, Jurij F Tasič, and Andrej Košir. 2013. Predicting and detecting the relevant contextual information in a movie-recommender system. *Interacting with Computers*, 25(1):74–90.

Nelleke Oostdijk, Martin Reynaert, Véronique Hoste, and Ineke Schuurman. 2013. The construction of a 500-million-word reference corpus of contemporary written dutch. In *Essential speech and language technology for Dutch*, pages 219–247. Springer, Berlin, Heidelberg.

Roeland Ordelman, Franciska de Jong, Arjan Van Hessen, and Hendri Hondorp. 2007. Twnc: a multifaceted dutch news corpus. *ELRA Newsletter*, 12(3/4):4–7.

Eli Pariser. 2011. *The filter bubble: How the new personalized web is changing what we read and how we think*. Penguin.

Siddharth Patwardhan and Ellen Riloff. 2009. A unified model of phrasal and sentential evidence for information extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 151–160.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Jakub Piskorski, Jacek Haneczok, and Guillaume Jacquet. 2020. New benchmark corpus and models for fine-grained event classification: To bert or not to bert? In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6663–6678.

Kim Schouten, Philip Ruijgrok, Jethro Borsje, Flavius Frasincar, Leonard Levering, and Frederik Hogenboom. 2010. A semantic web-based approach for personalizing news. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 854–861.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*.

Hans Smessaert and Liesbeth Augustinus. 2010. Nederbooms. *linguistics*, 2012.

Zhiyi Song, Ann Bies, Stephanie Strassel, Tom Riese, Justin Mott, Joe Ellis, Jonathan Wright, Seth Kulick, Neville Ryant, and Xiaoyi Ma. 2015. From light to rich ere: annotation of entities, relations, and events. In *Proceedings of the the 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*. Leibniz-Institut für Deutsche Sprache.

Stéphan Tulkens, Chris Emmery, and Walter Daelemans. 2016. Evaluating unsupervised dutch word embeddings as a linguistic resource. *arXiv preprint arXiv:1607.00225*.

Marco A Valenzuela-Escárcega, Gus Hahn-Powell, Mihai Surdeanu, and Thomas Hicks. 2015. A domain-independent rule-based framework for event extraction. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 127–132.

Gertjan Van Noord et al. 2006. At last parsing is now operational. In *TALN06. Verbum Ex Machina. Actes de la 13e conference sur le traitement automatique des langues naturelles*, pages 20–42.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Jesse Vig. 2019. A multiscale visualization of attention in the transformer model. *arXiv preprint arXiv:1906.05714*.

Jesse Vig and Yonatan Belinkov. 2019. Analyzing the structure of attention in a transformer language model. *arXiv preprint arXiv:1906.04284*.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. Bertje: A dutch bert model. *arXiv preprint arXiv:1912.09582*.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. *arXiv preprint arXiv:1909.03546*.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*, 57:45.

Boyi Xie, Rebecca Passonneau, Leon Wu, and Germán G Creamer. 2013. Semantic frames to predict stock price movement. In *Proceedings of the 51st annual meeting of the association for computational linguistics*, pages 873–883.