Making History Count

Gabriele Cerizza

Università degli Studi di Milano
gabriele.cerizza@studenti.unimi.it
https://github.com/gabrielecerizza/information_retrieval_project

1 Introduction

In this report we detail our findings in the study of two tasks related to project number 9 for the Information Retrieval course of University of Milan[1]: detection of shifts in the meaning of words across time (Section 2) and extraction of historical events from text (Section 3).

2 Semantic shifts

The first task is concerned with measuring, in a data-driven way, the diachronic semantic shift or lexical semantic change (LSC) that affected words across time. Formally, given a set of words W and two time periods t_1 and t_2, we want to measure

$$D(w_{t_1}, w_{t_2}) = \text{distance}(w_{t_1}, w_{t_2}) \,, \tag{1}$$

where w_{t_1} and w_{t_2} correspond to the same word w used in corpora of time periods t_1 and t_2.

We proceed to review known approaches to tackle the problem and then we discuss three possible strategies to measure LSC by exploiting word embeddings originated from a corpus of historical documents and from a corpus of contemporary documents.

2.1 Related work

In literature, the attempts to capture semantic shifts by way of word embeddings can be categorized in mainly two families: static embeddings or type-based models, and contextualized embeddings or token-based models.

Static embeddings models Static embeddings represent each word with a single vector. This vector can be considered a summarization of the occurrences of a word in different contexts. Popular static embeddings are obtained from skip-gram with negative sampling (SGNS) [26], GloVe [29] and fastText [2].

[1] https://island.ricerca.di.unimi.it/~alfio/shared/inforet/2020-21/inforet-projects.html

Static embeddings can be exploited to detect semantic shifts in a straightforward way. We train two models on corpora of different time periods and then we measure the cosine distance between the embeddings of a given word obtained from the two models [13]. The issue in this direct comparison is that SGNS and the other neural models are stochastic in nature and therefore they produce embeddings that may be differently rotated along the axes (invariance under rotation). Common solutions include:

- orthogonal Procrustes method, which aligns the embeddings to the same coordinate axes [13], but may introduce noise in the projections [9];
- initialization of the weights of one model with the weights of the other model, which can be problematic for new words [15,35];
- second-order similarity, which first computes the similarity of a word to other words in one vector space, then computes the similarity of the same word to other words in the other vector space, and finally compares the two similarities [15,35].

Contextualized embeddings models Recently developed neural models like BERT [7] generate different word embeddings for each different context in which a word appears. The pretrained contextualized embeddings are sometimes employed by the authors as is, without fine-tuning on the historical corpus [31,17].

Contextualized embeddings models start by extracting a vector for each context in which a word appears. A context is usually a sentence (or a portion thereof) in the two corpora. Then, these models cluster the vectors of the two periods to find the different senses of a word. After that, they measure how the frequency of the senses changed between the time periods to get an estimate of the semantic shift [12,31]. An alternative approach computes the average pairwise distance between each vector of one period and each vector of the other period [17].

2.2 Proposed methods

We explored three methods to detect semantic shifts. Two are based on static embeddings and a third one on contextualized embeddings.

Orthogonal Procrustes method (OP) We started by collecting pretrained static word embeddings. For the historical corpus we used the embeddings provided in [34]. These embeddings were trained with fastText on documents dating from 1860 to 1939 and taken from the Corpus of Historical American English (COHA[2]) for a total of 198M tokens. For the contemporary corpus we used fastText word embeddings trained on Wikipedia and news[3] for a total of 16B tokens and 1M word vectors.

[2] https://corpus.byu.edu/coha/
[3] https://fasttext.cc/docs/en/english-vectors.html

Considering the size of the contemporary vocabulary, as well as the fact that it contained misspelled words, we decided to analyze only the 5000 most frequent words in the contemporary vocabulary, intersected with the historical vocabulary. We also made sure to analyze the "target" words indicated by SemEval-2020 [32] for evaluation purposes (see Subsection 2.3).

Finally, we aligned the vectors with orthogonal Procrustes and computed the cosine distance between the two embeddings of each selected word.

Nearest neighbors method (NN) This approach leverages ideas taken from the second-order similarity techniques used with static embeddings. We used the same historical and contemporary word embeddings described for OP. Likewise, we analyzed the same words selected for OP.

The proposed second-order similarity is computed as follows. For each selected word we take the 15 nearest neighbors in the historical vector space; then we measure the cosine distance between the word and these neighbors in the historical vector space; after that, we measure the cosine distance between the word and the same neighbors, but this time in the contemporary vector space; finally we compute the mean squared error (MSE) between the distances measured in the historical vector space and the distances measured in the contemporary vector space. Then we do the same for the 15 nearest neighbors in the contemporary model. Finally, we take the mean between the MSE measured on the historical vector space neighbors and the MSE measured on the contemporary vector space neighbors.

Given this second-order similarity, we decided to shrink the vocabulary of the two models to their intersection. In this way we guaranteed that each neighbor found in one vector space was also present in the other vector space.

Jensen-Shannon distance method (JSD) This method is based on contextualized embeddings. The corpora were taken from SemEval-2020 [32]. The historical corpus contained shuffled sentences from 1810-1860, while the contemporary corpus contained shuffled sentences from 1960-2010. Both corpora were composed of 6M tokens.

We analyzed the 5000 most frequent words in both corpora, keeping only adjectives, nouns and verbs and filtering stop words, punctuations and tokens containing non-alphabetic characters. We aggregated the embeddings according to lemma and POS tags. Our assumption is that the semantic shift of a word is invariant to declension and conjugation: "cat" and "cats" should be subject to the same semantic shift; likewise for "write" and "wrote". Note that in the previous methods we were not able to distinguish words based on POS, since we only had the embeddings, thus the noun "attack" and the verb "attack" had the same embeddings.

We generated the embeddings from BERT, using the "bert-base-cased"[4] pretrained weights, by feeding the transformer with sentences from the corpora. We

[4] https://huggingface.co/bert-base-cased

fine-tuned the model on the historical corpus for 5 epochs. Note that we used only one model to generate embeddings for both time periods. Since BERT generates different embeddings for different contexts, it suffices that BERT acquired knowledge of the contexts used in both time periods [24]. For the embeddings, we took only the hidden state of the last layer, which is reportedly the layer most related to semantics [17].

Following the approach described in [31], we first employed an autoencoder to reduce the dimensionality of the embeddings from 768 to 20. Then, the dimensionality was further reduced to 10 with UMAP. The combination of autoencoder and UMAP was shown to be effective for clustering in [25]. We clustered the embeddings of a given word with HDBSCAN, which is a variant of DBSCAN with improved robustness and a single hyperparameter [3].

Interpreting the clusters as senses of a word, we measured the frequency of each sense in the two time periods. Finally, we used the Jensen-Shannon distance to measure the similarity between the two distributions.

2.3 Results

In Table 1 we show the 10 words whose semantic shift was most prominent according to the proposed methods. In OP we can see many words whose meaning changed due to technological evolution, like "cd", "tv" and "bot". OP and NN also capture shifts of historical or cultural nature, like "isis" and "gay". We observe that the International System of Units was introduced in 1960, hence a possible explanation for the shift in "km". The semantic shifts detected by JSD are not as readily decipherable.

In Figure 1 we give an example of how the shift in senses was perceived by the JSD method. We can see, for instance, that the verb "score" assumed a meaning more related to sports as the time passed.

Table 1. Top 10 words with the most semantic shift detected by the methods.

OP	NN	JSD
cd	deletion	negro (adj.)
romney	km	people (noun)
km	gay	golf (noun)
diff	diff	shopping (noun)
deletion	outstanding	overall (adj.)
tv	implement	businessman (noun)
template	highlight	switch (verb)
isis	parameter	investor (noun)
bot	red	motor (noun)
highlight	template	user (noun)

The instructions provided for project number 9 stated that the evaluation of the results would be carried out by experts in the linguistics and history

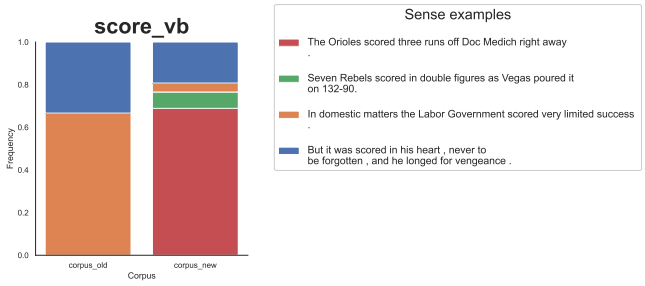

Fig. 1. Shift in the frequency of senses for the verb "score".

domains. To this end, detailed results for each method can be perused in our repository.

Furthermore, we evaluated our results against a ground truth for the semantic shift occurred in 37 words provided by SemEval-2020 [32]. In Table 2 we compare the Spearman's rank-order correlation of our methods with the top 5 systems in the SemEval-2020 shared task. We deduce that our methods are competitive with other state-of-the-art systems.

Table 2. Spearman's rank-order correlation to SemEval-2020 ground truth for the proposed methods and the top systems in the competition.

OP	NN	JSD	UG_Student_Intern	Jiaxin & Jinan	cs2020	UWB	Discovery_Team
.391	.341	.365	**.422**	.325	.375	.367	.361

2.4 Conclusion

Tables 1 and 2 suggest that, on the LSC task, static embeddings methods perform slightly better than contextualized embeddings methods. The same conclusion can be found in literature [17,32]. Possible reasons for this behavior are the small size of the corpora for the JSD method and a need of further fine-tuning. Another possible reason lies in the fact that the historical corpus contains a lot of artifacts and wrongly tokenized words.

Apart from these issues, a possible improvement in the JSD method involves tuning the HDBSCAN hyperparameter to find a number of senses for each word that is similar to the number of word synsets found in WordNet.

3 Historical events extraction

The second task mentioned in project number 9 involves the detection of historical events in a text and the extraction of the event components, such as

dates, historical figures and locations. The Histo Corpus described in [34] was suggested for this task. The event recognition model was to be evaluated on Wikipedia pages.

3.1 Definitions

The notion of "historical event" is not clearly defined. Since "individuation criteria are not given by nature or language" [33], ultimately the choice of what constitutes an historical event is arbitrary. In literature, events were labeled as historical when they involved conflicts [6]. Here we consider historical such events and entities that may be the subject of a paragraph in a history textbook. These primarily concern military and political occurrences.

3.2 Data sets

The Histo Corpus is composed of news and travel narratives from 1865-1926 and is annotated with events, which mainly consist in verbs and participles. Three issues prevented us from using this corpus.

1. The corpus is "historical" in the sense that the documents are not contemporary, but the annotated events are not "historical" in the sense that they are not characterized by historical figures or by episodes that we may find in a history textbook.
2. Since the corpus is annotated with "common" events described using a "historical" language, we would have no way to recognize "historical" events described using a "contemporary" language in Wikipedia.
3. The annotations are limited to the events themselves and do not include event arguments, like dates and people involved in the event.

To the best of our knowledge, there are no public English datasets in which historical events are annotated along with their arguments. An interesting dataset would be the one described in [16], in which documents from 1827-1909 are annotated with events related to conflicts and law, along with their arguments. However, this dataset has not yet been disclosed as of the time of this writing.

For the recognition of events and their arguments, we employed the RAMS dataset [10], in which 139 event types, each with up to 5 arguments, were annotated. Each document of the dataset contains a single event. The peculiarity of this dataset is that the arguments of an event may be found in a different sentence from the one in which the event is mentioned. This makes the task arguably more difficult compared to more well-known datasets, such as ACE 2005[5].

We also built a dataset composed of 1024 pages taken from Wikipedia, for a total of approximately 4M tokens. Each page was classified as historical or not based on the properties of the corresponding Wikidata entry. Each page

[5] https://catalog.ldc.upenn.edu/LDC2006T06

was split into paragraphs to which we assigned the same class of the page, resulting in 16075 historical paragraphs and 23673 non-historical paragraphs. Finally, each paragraph was annotated in the BIO format with the Wikipedia entities mentioned therein, by exploiting the links in the text. Those entities were likewise classified as historical or not based on their Wikidata entry.

3.3 Strategy

In tackling the task, we adopted the following strategy. First, we built a model to classify a given paragraph as historical or not. We make the assumption that Wikipedia pages related to historical entities are more likely to describe historical events in their paragraphs. Note that we cannot base our classification solely on the entities mentioned in a paragraph, because: (i) the link to an entity is usually provided only for the first mention and thus only the first mention of an entity can be annotated; (ii) the surface form of an entity can vary greatly (e.g., the text "up the toe of Italy" contains a link to the "Italian campaign (World War II)" entity in the Wikipedia page for Bernard Montgomery).

Once the paragraphs related to historical events were identified, a second model was employed to extract the event and the pertaining arguments.

3.4 Related work

In this subsection we briefly outline the models proposed in literature for event and argument extraction. Formally, the problem can be stated as follows. We define an ontology of event types $t \in T$, each of which is associated with a set of n_t arguments A_t. Given a document $d = \langle w_1, \ldots, w_m \rangle$, we group the m words of the document in z spans $s = \langle w_k, \ldots, w_l \rangle \in S$. The problem is then to find which span s (also called "trigger") corresponds to an event t, if any, and which spans correspond to the n_t arguments of the event t, if any.

We categorize the models found in literature in two families: span classification approaches and machine reading comprehension (MRC) approaches. We did not consider approaches that exploited external resources, like the Wikipedia and DBpedia ontologies. One such example can be found in [1].

Span classification approaches These approaches exploit event and argument annotations to assign labels to each span. There are methods that learn embeddings for each span, possibly adding external features such as POS and dependencies in the parser tree, and then use a neural network [10,40,28] or transformers [5] to classify each span. Some methods also employ graphs within the process [20,36,23,27]. Most methods extract events and arguments jointly, since pipelining the two tasks is believed to propagate and amplify errors. However, recent approaches are challenging this assumption [40].

MRC approaches These approaches do not label each span, but rather produce a natural language response to a natural language prompt. These approaches display improved performance in few-shot and zero-shot settings compared to classification approaches. There are models based on slot-filling methods that fill placeholders in a predefined template with spans extracted from the text [4,19]. Other models treat the problem as a question answering task [8,11,21] or as a textual entailment task [11]. Other models yet use sequence-to-structure text generation [22].

3.5 Proposed methods

Paragraph classification We adopted a multi-task learning model (MTL) to classify paragraphs as historical or not. First, we obtained BERT embeddings for each token. Then, we used two feed-forward neural networks (FFNN) to jointly learn (i) the paragraph class, and (ii) the tag of the entities referenced in the paragraph. Cross-entropy loss was computed separately for each task and then summed according to learned weights [14].

We compared our model with a BiLSTM classifier and a BERT classifier as baseline methods.

Event and argument extraction To identify trigger words and thus extract events, we followed the span classification approach described in [40]. After enumerating all spans up to 3 tokens long, we obtained embeddings for each span by aggregating the BERT embeddings of the tokens. Then, we passed the embeddings through a FFNN to classify each span either as an event type t or as a non-event. We used cross-entropy loss.

For argument extraction, we followed the MRC approach described in [19,38]. We fed a BART [18] model with the paragraph and an event template with placeholders "⟨arg⟩" for each argument. After training, the model learned to generate the same template with the placeholders filled with text spans. We restricted the language vocabulary to that of the input during generation. We used cross-entropy loss on the logits obtained from the masked language modeling training.

Event and argument extraction can be performed jointly in a single model by aggregating their losses. We kept the two tasks separated due to hardware constraints and provided the argument model with the gold event type.

We compared our model with results reported in literature.

3.6 Results

In Table 3 we show the results for the paragraph classification task. Table 4 compares the performances on the event and argument extraction tasks. Note that the authors employed different training settings, such as using gold argument spans and gold event types or considering the syntactical head-words (most representative tokens) of the arguments instead of the whole spans, making it difficult to directly compare the models.

Table 3. Accuracy and F1-score of the models for the paragraph classification task.

Model	Accuracy	F1-score
MTL	**0.854**	**0.716**
BiLSTM	0.794	0.634
BERT	0.799	0.630

Table 4. Precision (P), recall (R) and F1-score of different models on the official test set of the RAMS data set. TCD stands for type-constrained decoding and refers to the practice of considering only the top-scoring n_t arguments for an event type t with n_t arguments. For our model we reported the micro-average scores.

	Gold Arg. Spans	Events			Arguments		
Model		P	R	F1	P	R	F1
Ebner et al. [10]	Yes	-	-	-	62.8	**74.9**	68.3
Ebner et al. (TCD) [10]	Yes	-	-	-	78.1	69.2	73.3
Zhang et al. [39]	Yes	-	-	-	71.5	66.2	68.8
Zhang et al. (TCD) [39]	Yes	-	-	-	81.1	66.2	73.0
Wei et al. [37]	Yes	-	-	-	**82.0**	71.6	**76.6**
Li et al. [19]	No	-	-	-	-	-	48.6
Zhang et al. [39]	No	-	-	-	-	-	40.1
Zhang et al. (TCD) [39]	No	-	-	-	-	-	41.8
Wen et al. [38]	No	-	-	-	-	-	48.6
Wei et al. [37]	No	-	-	-	53.1	**42.7**	47.4
Lai et al. [16]	No	**71.9**	74.7	**73.2**	-	-	-
Pouran Ben Veyseh et al. [30]	No	55.5	**78.6**	65.1	-	-	-
Our model	No	0.0290	0.0287	0.0289	**62.7**	42.0	**50.3**

3.7 Conclusion

With regard to the paragraph classification task, our MTL model significantly outperforms the baseline methods. The performance could be further improved with a more fine-grained analysis of the Wikidata properties of each Wikipedia page.

Our event model struggles to correctly identify trigger words. One obvious reason is that the vast majority of the spans do not contain events and, therefore, finding not only the correct event span but also the correct event type, from a total of 139 types, becomes a highly imbalanced problem. The results can also be ascribed to the complexity of the data set, whose documents comprise multiple sentences. Moreover, the data set was intended exclusively for argument linking. Finally, trigger words are not limited to verbs, which is at odds with most of the other event detection data sets.

Our argument model compares favourably with the baseline methods. Note that our model does not know the gold argument spans. Indeed, since our model generates text rather than classify spans, it disregards spans altogether.

References

1. Ahonen, E., Hyvonen, E.: Publishing historical texts on the semantic web - a case study. In: 2009 IEEE International Conference on Semantic Computing. pp. 167–173 (2009). https://doi.org/10.1109/ICSC.2009.9
2. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5**, 135–146 (2017)
3. Campello, R.J.G.B., Moulavi, D., Sander, J.: Density-based clustering based on hierarchical density estimates. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (eds.) Advances in Knowledge Discovery and Data Mining. pp. 160–172. Springer Berlin Heidelberg, Berlin, Heidelberg (2013)
4. Chen, Y., Chen, T., Ebner, S., White, A.S., Van Durme, B.: Reading the manual: Event extraction as definition comprehension. In: Proceedings of the Fourth Workshop on Structured Prediction for NLP. pp. 74–83. Association for Computational Linguistics, Online (Nov 2020). https://doi.org/10.18653/v1/2020.spnlp-1.9, https://aclanthology.org/2020.spnlp-1.9
5. Chen, Y., Chen, T., Van Durme, B.: Joint modeling of arguments for event understanding. In: Proceedings of the First Workshop on Computational Approaches to Discourse. pp. 96–101. Association for Computational Linguistics, Online (Nov 2020). https://doi.org/10.18653/v1/2020.codi-1.10, https://aclanthology.org/2020.codi-1.10
6. Cybulska, A.K., Vossen, P.: Historical event extraction from text. In: Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities. pp. 39–43. Association for Computational Linguistics, Portland, OR, USA (Jun 2011), https://aclanthology.org/W11-1506
7. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR (2018), http://arxiv.org/abs/1810.04805

8. Du, X., Cardie, C.: Event extraction by answering (almost) natural questions. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 671–683. Association for Computational Linguistics, Online (Nov 2020). https://doi.org/10.18653/v1/2020.emnlp-main.49, https://aclanthology.org/2020.emnlp-main.49
9. Dubossarsky, H., Hengchen, S., Tahmasebi, N., Schlechtweg, D.: Time-out: Temporal referencing for robust modeling of lexical semantic change. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 457–470. Association for Computational Linguistics, Florence, Italy (Jul 2019). https://doi.org/10.18653/v1/P19-1044, https://aclanthology.org/P19-1044
10. Ebner, S., Xia, P., Culkin, R., Rawlins, K., Van Durme, B.: Multi-sentence argument linking. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 8057–8077. Association for Computational Linguistics, Online (Jul 2020). https://doi.org/10.18653/v1/2020.acl-main.718, https://aclanthology.org/2020.acl-main.718
11. Feng, R., Yuan, J., Zhang, C.: Probing and fine-tuning reading comprehension models for few-shot event extraction. CoRR **abs/2010.11325** (2020), https://arxiv.org/abs/2010.11325
12. Giulianelli, M., Del Tredici, M., Fernández, R.: Analysing lexical semantic change with contextualised word representations. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 3960–3973. Association for Computational Linguistics, Online (Jul 2020). https://doi.org/10.18653/v1/2020.acl-main.365, https://aclanthology.org/2020.acl-main.365
13. Hamilton, W.L., Leskovec, J., Jurafsky, D.: Diachronic word embeddings reveal statistical laws of semantic change. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1489–1501. Association for Computational Linguistics (Aug 2016)
14. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. CoRR **abs/1705.07115** (2017), http://arxiv.org/abs/1705.07115
15. Kutuzov, A., Øvrelid, L., Szymanski, T., Velldal, E.: Diachronic word embeddings and semantic shifts: a survey. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 1384–1397. Association for Computational Linguistics (Aug 2018)
16. Lai, V., Nguyen, M.V., Kaufman, H., Nguyen, T.H.: Event extraction from historical texts: A new dataset for black rebellions. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. pp. 2390–2400. Association for Computational Linguistics, Online (Aug 2021). https://doi.org/10.18653/v1/2021.findings-acl.211, https://aclanthology.org/2021.findings-acl.211
17. Laicher, S., Kurtyigit, S., Schlechtweg, D., Kuhn, J., im Walde, S.S.: Explaining and improving BERT performance on lexical semantic change detection. CoRR (2021), https://arxiv.org/abs/2103.07259
18. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7871–7880. Association for Computational Linguistics, Online (Jul

2020). https://doi.org/10.18653/v1/2020.acl-main.703, https://aclanthology.org/2020.acl-main.703
19. Li, S., Ji, H., Han, J.: Document-level event argument extraction by conditional generation. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 894–908. Association for Computational Linguistics, Online (Jun 2021). https://doi.org/10.18653/v1/2021.naacl-main.69, https://aclanthology.org/2021.naacl-main.69
20. Lin, Y., Ji, H., Huang, F., Wu, L.: A joint neural model for information extraction with global features. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7999–8009. Association for Computational Linguistics, Online (Jul 2020). https://doi.org/10.18653/v1/2020.acl-main.713, https://aclanthology.org/2020.acl-main.713
21. Liu, J., Chen, Y., Liu, K., Bi, W., Liu, X.: Event extraction as machine reading comprehension. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1641–1651. Association for Computational Linguistics, Online (Nov 2020). https://doi.org/10.18653/v1/2020.emnlp-main.128, https://aclanthology.org/2020.emnlp-main.128
22. Lu, Y., Lin, H., Xu, J., Han, X., Tang, J., Li, A., Sun, L., Liao, M., Chen, S.: Text2event: Controllable sequence-to-structure generation for end-to-end event extraction. CoRR **abs/2106.09232** (2021), https://arxiv.org/abs/2106.09232
23. Luan, Y., Wadden, D., He, L., Shah, A., Ostendorf, M., Hajishirzi, H.: A general framework for information extraction using dynamic span graphs. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 3036–3046. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019). https://doi.org/10.18653/v1/N19-1308, https://aclanthology.org/N19-1308
24. Martinc, M., Kralj Novak, P., Pollak, S.: Leveraging contextual embeddings for detecting diachronic semantic shift. In: Proceedings of the 12th Language Resources and Evaluation Conference. pp. 4811–4819. European Language Resources Association (May 2020)
25. McConville, R., Santos-Rodríguez, R., Piechocki, R.J., Craddock, I.: N2D: (not too) deep clustering via clustering the local manifold of an autoencoded embedding. CoRR (2019), http://arxiv.org/abs/1908.05968
26. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems. vol. 26. Curran Associates, Inc. (2013)
27. Nguyen, M.V., Lai, V.D., Nguyen, T.H.: Cross-task instance representation interactions and label dependencies for joint information extraction with graph convolutional networks. CoRR **abs/2103.09330** (2021), https://arxiv.org/abs/2103.09330
28. Nguyen, T.M., Nguyen, T.H.: One for all: Neural joint modeling of entities and events. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 6851–6858 (2019)
29. Pennington, J., Socher, R., Manning, C.: GloVe: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543. Association for Computational Linguistics (Oct 2014)

30. Pouran Ben Veyseh, A., Lai, V., Dernoncourt, F., Nguyen, T.H.: Unleash GPT-2 power for event detection. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 6271–6282. Association for Computational Linguistics, Online (Aug 2021). https://doi.org/10.18653/v1/2021.acl-long.490, https://aclanthology.org/2021.acl-long.490
31. Rother, D., Haider, T., Eger, S.: CMCE at SemEval-2020 task 1: Clustering on manifolds of contextualized embeddings to detect historical meaning shifts. In: Proceedings of the Fourteenth Workshop on Semantic Evaluation. pp. 187–193. International Committee for Computational Linguistics, Barcelona (online) (Dec 2020), https://aclanthology.org/2020.semeval-1.22
32. Schlechtweg, D., McGillivray, B., Hengchen, S., Dubossarsky, H., Tahmasebi, N.: SemEval-2020 task 1: Unsupervised lexical semantic change detection. In: Proceedings of the Fourteenth Workshop on Semantic Evaluation. pp. 1–23. International Committee for Computational Linguistics, Barcelona (online) (Dec 2020), https://aclanthology.org/2020.semeval-1.1
33. Shaw, R.: Events and Periods as Concepts for Organizing Historical Knowledge. Ph.D. thesis, University of California, Berkeley (01 2010)
34. Sprugnoli, R., Tonelli, S.: Novel Event Detection and Classification for Historical Texts. Computational Linguistics **45**(2), 229–265 (06 2019)
35. Tahmasebi, N., Borin, L., Jatowt, A.: Survey of computational approaches to diachronic conceptual change. CoRR (2018), http://arxiv.org/abs/1811.06278
36. Wadden, D., Wennberg, U., Luan, Y., Hajishirzi, H.: Entity, relation, and event extraction with contextualized span representations. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 5784–5789. Association for Computational Linguistics, Hong Kong, China (Nov 2019). https://doi.org/10.18653/v1/D19-1585, https://aclanthology.org/D19-1585
37. Wei, K., Sun, X., Zhang, Z., Zhang, J., Zhi, G., Jin, L.: Trigger is not sufficient: Exploiting frame-aware knowledge for implicit event argument extraction. In: ACL 2021: 59th annual meeting of the Association for Computational Linguistics. pp. 4672–4682 (2021)
38. Wen, H., Lin, Y., Lai, T., Pan, X., Li, S., Lin, X., Zhou, B., Li, M., Wang, H., Zhang, H., Yu, X., Dong, A., Wang, Z., Fung, Y., Mishra, P., Lyu, Q., Surís, D., Chen, B., Brown, S.W., Palmer, M., Callison-Burch, C., Vondrick, C., Han, J., Roth, D., Chang, S.F., Ji, H.: Resin: A dockerized schema-guided cross-document cross-lingual cross-media information extraction and event tracking system. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations. pp. 133–143 (2021)
39. Zhang, Z., Kong, X., Liu, Z., Ma, X., Hovy, E.: A two-step approach for implicit event argument detection. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7479–7485. Association for Computational Linguistics, Online (Jul 2020). https://doi.org/10.18653/v1/2020.acl-main.667, https://aclanthology.org/2020.acl-main.667
40. Zhong, Z., Chen, D.: A frustratingly easy approach for entity and relation extraction. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 50–61. Association for Computational Linguistics, Online (Jun

2021). https://doi.org/10.18653/v1/2021.naacl-main.5, https://aclanthology.org/2021.naacl-main.5