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 Università degli Studi di Milano.[1]

2 Semantic Shifts

The first task is concerned with measuring the diachronic semantic shift or lexical semantic change (LSC) of words across time. Formally, given a set of words W, a word embedding function E and two time periods t_1 and t_2, we want to measure

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

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.

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) [27], GloVe [30] and fastText [1].

We could train two models on corpora of different time periods and then measure the cosine distance between the embeddings of a given word. The issue in this direct comparison is that SGNS and the other neural models are stochastic in nature and may produce embeddings that are differently rotated along the axes (invariance under rotation). Common solutions include:

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

- orthogonal Procrustes method, which aligns the embeddings to the same coordinate axes [12,33], but may introduce noise in the projections [8,45];
- initialization of the weights of one model with the weights of the other model, which can be problematic for new words [15,39];
- 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 [15,39].

Contextualized Embeddings Models. Recently developed neural models like BERT [6] generate different word embeddings for each different context (sentence) in which a word appears.

Contextualized embeddings models start by extracting a vector for each occurrence of a word. 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 [11,34]. An alternative approach computes the average pairwise distance between each vector of one period and each vector of the other period [14,17,31].

2.2 Proposed Methods

Orthogonal Procrustes Method (OP). We started by collecting pretrained static word embeddings. For the historical corpus we used the embeddings provided in [38]. These embeddings were trained with fastText on documents dated 1860-1939 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.

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 included the "target" words indicated by SemEval-2020 [35] 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 over static embeddings. We used the same word embeddings and selected the same words as described for OP.

For each selected word we took the 15 nearest neighbors in the historical vector space; then we measured the cosine distance between the word and these neighbors in the historical vector space; after that, we measured the cosine distance between the word and the same neighbors, but in the contemporary vector

[2] https://corpus.byu.edu/coha/

[3] https://fasttext.cc/docs/en/english-vectors.html

space; finally we computed 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 did the same for the 15 nearest neighbors in the contemporary model. Finally, we took the mean between the two MSE.

We shrunk the vocabulary of the two models to their intersection to guarantee that each neighbor found in one vector space was also present in the other.

Jensen-Shannon Distance Method (JSD). This method is based on contextualized embeddings. The corpora were taken from SemEval-2020 [35]. 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. We filtered stop words 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.

We generated the embeddings from BERT. We fine-tuned the transformer 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 [34], we first employed an autoencoder and UMAP [26] to reduce the dimensionality of the embeddings from 768 to 10. The combination of autoencoder and UMAP was shown to be effective for clustering in [25]. Then, we clustered the embeddings of a given word with HDBSCAN, which is a variant of DBSCAN with improved robustness and a single hyperparameter [2].

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 senses distributions.

2.3 Results

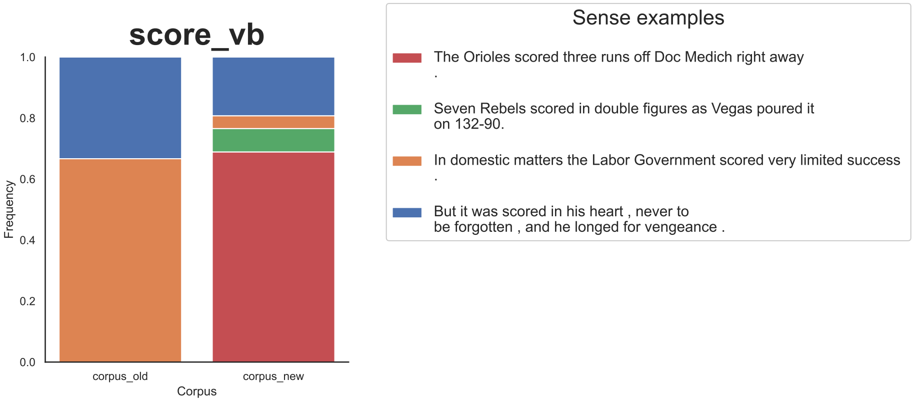
In Table 1 we show the 10 words with most semantic shift according to the proposed methods. OP identified many words whose meaning changed due to technological evolution, like “cd”, “tv” and “bot”. OP and NN also captured shifts of historical or cultural nature, like “isis” and “gay”. The semantic shifts 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.

Furthermore, we evaluated our results against a ground truth for the semantic shift occurred in 37 words provided by SemEval-2020 [35]. In Table 2 we compare

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)



Sense examples

🟥 The Orioles scored three runs off Doc Medich right away
.

🟩 Seven Rebels scored in double figures as Vegas poured it
on 132-90.

🟧 In domestic matters the Labor Government scored very limited success
.

🟦 But it was scored in his heart , never to
be forgotten , and he longed for vengeance .

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

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 static embeddings methods perform slightly better than contextualized embeddings methods. The same conclusion can be found in literature [17, 35]. 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.

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 Event Extraction

The second task involves the detection of historical events in a text and the extraction of the event arguments, such as dates, historical figures and locations. The Histo Corpus described in [38] was suggested for this task. The event recognition model was to be evaluated on Wikipedia pages.

3.1 Definition of Historical Event

The notion of "historical event" is not clearly defined. Since "individuation criteria are not given by nature or language" [36], ultimately the choice of what constitutes an historical event is arbitrary. 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, which is consistent with [5].

3.2 Data Sets

The Histo Corpus is composed of news and travel narratives from 1865-1926 and is annotated with events, consisting mainly 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 arguments, like dates and people involved in the events.

To the best of our knowledge, there are no public English data sets in which historical events are annotated along with their arguments. A promising data set, described in [16], is yet to be published as of the time of this writing.

For the recognition of events and their arguments, we employed the RAMS data set [9], in which 139 event types, each with up to 5 arguments, were annotated. Each document contains 5 sentences. The arguments of an event may be found in different sentences from the one in which the event is mentioned.

We also built a data set composed of 1024 pages taken from Wikipedia, for a total of 4M tokens. Each page was classified as historical or not based on the properties of the corresponding Wikidata entry. Each page was split into paragraphs to which we assigned the same class of the page. By exploiting the links in the text and Wikidata, we likewise classified the entities mentioned in each paragraph as historical or not. Finally, each paragraph was annotated in the BIO format with the Wikipedia entities mentioned therein.

3.3 Strategy

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; (ii) the surface form of an entity can vary greatly.

Once the paragraphs related to historical events were identified, other models were employed to extract the event and the pertaining arguments.

3.4 Related Work

Formally, the event extraction 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 want to find which event t, if any, is described in d, possibly identifying the event “trigger” word, and which words 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.

Span Classification Approaches. These approaches group words in spans and 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 tags and dependencies in the parser tree, and then use a neural network [9,29,44] or transformers [4] to classify each span. Some methods also employ graphs within the process [20,23,28,40].

MRC Approaches. These approaches do not label spans, but rather produce a natural language response to a natural language prompt. There are models that fill placeholders in a predefined template with spans extracted from the text [3,19]. Other models treat the problem as a question answering task [7,10,21] or as a textual entailment task [10]. 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 [13].

We compared our model with a BiLSTM classifier and a BERT classifier.

Event and Argument Extraction. To identify events, we followed the span classification approach described in [44]. 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 [19,42]. We fed a BART [18] model with the paragraph and an event template with an "⟨arg⟩" placeholder 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 tried the same approach also for document-level event identification (EventGen).

We were unable to perform event and argument extraction jointly due to hardware constraints, so we provided the argument model with the gold event type during training. We compared our model with results reported in literature and with BiLSTM and BERT document-level classifiers.

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 (Acc), precision (P), recall (R) and F1-score for the paragraph classification task on the test set. We reported weighted average scores.

Model	Acc	P	R	F1
MTL	**84.2**	**80.6**	**84.2**	**82.4**
BiLSTM	74.4	74.7	74.4	74.5
BERT	73.0	76.4	73.0	74.7

Table 4. Precision (P), recall (R) and F1-score 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 event models we reported the weighted average scores, while for the argument model we reported the metrics computed according to the official RAMS scorer.

	Gold Arg. Spans	Events			Arguments		
Model		P	R	F1	P	R	F1
Ebner et al. [9]	Yes	-	-	-	62.8	**74.9**	68.3
Ebner et al. (TCD) [9]	Yes	-	-	-	78.1	69.2	73.3
Zhang et al. [43]	Yes	-	-	-	71.5	66.2	68.8
Zhang et al. (TCD) [43]	Yes	-	-	-	81.1	66.2	73.0
Wei et al. [41]	Yes	-	-	-	**82.0**	71.6	**76.6**
Li et al. [19]	No	-	-	-	-	-	48.6
Zhang et al. [43]	No	-	-	-	-	-	40.1
Zhang et al. (TCD) [43]	No	-	-	-	-	-	41.8
Wen et al. [42]	No	-	-	-	-	-	48.6
Wei et al. [41]	No	-	-	-	53.1	**42.7**	47.4
Lai et al. [16]	No	**71.9**	74.7	**73.2**	-	-	-
Pouran Ben Veyseh et al. [32]	No	55.5	**78.6**	65.1	-	-	-
Our BiLSTM baseline	No	0.1	2.3	0.2	-	-	-
Our BERT baseline	No	1.1	1.7	1.1	-	-	-
Our model (EventGen)	No	29.8	27.9	24.8	-	-	-
Our model	No	2.2	2.9	2.3	**62.7**	42.0	**50.3**

3.7 Conclusion

With regard to the paragraph classification task, our MTL model outperforms the baseline methods. Further improvement can be attained with a more fine-grained analysis of the Wikidata properties of each Wikipedia page.

Our event model struggles to correctly identify trigger words. One reason is that the vast majority of the spans do not contain events and, therefore, finding the correct event span and the correct event type becomes a highly imbalanced problem. The EventGen model fares better and the predictions show that in many cases the mistakes were semantically close to the gold labels. Our argument model compares favourably with the baseline methods. We could improve the model by appending argument type constraints to the templates [19, 37].

References

1. 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)
2. 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)
3. 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)
4. 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)
5. 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)
6. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR (2018)
7. 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)
8. 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)
9. 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)
10. Feng, R., Yuan, J., Zhang, C.: Probing and fine-tuning reading comprehension models for few-shot event extraction. CoRR (2020)
11. 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)
12. 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)
13. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. CoRR (2017)
14. Kutuzov, A., Giulianelli, M.: UiO-UvA at SemEval-2020 task 1: Contextualised embeddings for lexical semantic change detection. In: Proceedings of the Fourteenth Workshop on Semantic Evaluation. pp. 126–134. International Committee for Computational Linguistics, Barcelona (online) (Dec 2020)

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)
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)
18. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., et al.: 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)
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)
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)
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)
22. Lu, Y., Lin, H., Xu, J., Han, X., Tang, J., Li, A., et al.: Text2event: Controllable sequence-to-structure generation for end-to-end event extraction. CoRR (2021)
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)
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)
26. McInnes, L., Healy, J., Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction (2020)
27. 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)

28. 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 (2021)
29. 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)
30. 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)
31. Pömsl, M., Lyapin, R.: CIRCE at SemEval-2020 task 1: Ensembling context-free and context-dependent word representations. In: Proceedings of the Fourteenth Workshop on Semantic Evaluation. pp. 180–186. International Committee for Computational Linguistics, Barcelona (online) (Dec 2020)
32. 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)
33. Pražák, O., Přibáň, P., Taylor, S., Sido, J.: UWB at SemEval-2020 task 1: Lexical semantic change detection. In: Proceedings of the Fourteenth Workshop on Semantic Evaluation. pp. 246–254. International Committee for Computational Linguistics, Barcelona (online) (Dec 2020)
34. 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)
35. 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)
36. Shaw, R.: Events and Periods as Concepts for Organizing Historical Knowledge. Ph.D. thesis, University of California, Berkeley (01 2010)
37. Shwartz, V., West, P., Le Bras, R., Bhagavatula, C., Choi, Y.: Unsupervised commonsense question answering with self-talk. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 4615–4629. Association for Computational Linguistics, Online (Nov 2020)
38. Sprugnoli, R., Tonelli, S.: Novel Event Detection and Classification for Historical Texts. Computational Linguistics **45**(2), 229–265 (06 2019)
39. Tahmasebi, N., Borin, L., Jatowt, A.: Survey of computational approaches to diachronic conceptual change. CoRR (2018)
40. 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)
41. 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)

42. Wen, H., Lin, Y., Lai, T., Pan, X., Li, S., Lin, X., et al.: 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)
43. 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)
44. 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)
45. Zhou, J., Li, J.: TemporalTeller at SemEval-2020 task 1: Unsupervised lexical semantic change detection with temporal referencing. In: Proceedings of the Fourteenth Workshop on Semantic Evaluation. pp. 222–231. International Committee for Computational Linguistics, Barcelona (online) (Dec 2020)