Proceedings of the 18th Conference on Computer Science and Intelligence Systems pp. 1041–1046
DOI: 10.15439/2023F4827
ISSN 2300-5963 ACSIS, Vol. 35

Mechanism for detecting cause-and-effect relationships in court judgments

Łukasz Kurant
0000-0002-2523-5952
University of Maria Curie-Sklodowska
in Lublin
Pl. M. Curie-Skłodowskiej 5, 20-031 Lublin, Poland
Email: lukasz.kurant@mail.umcs.pl

***Abstract*—Among the solutions for the detection of cause-and-effect relationships are methods based on knowledge, statistical solutions or methods allowing the use of deep learning. The solution presented in the article uses bidirectional artificial neural networks LSTM to detect such relationships in legal texts in Polish. The analysis was performed at the sentence level, but due to the specific legal language and the focus on Polish, two separated networks were used in the experiment. The task of the first one is to classify whether a sentence contains a conditional, while the second one is to identify the elements of this relationship. Both use word embedding sets for the Polish language corpus. The results of the experiment prove that it is possible to perform such extraction with satisfactory results, and raise questions and point to further possible ways forward.**

I. Introduction

DETECTING cause-and-effect relationships in texts, is a task that requires advanced cognitive processes and is not a trivial problem. Inference itself can often be a very difficult task for human beings, so it is not surprising that attempts are made to automatically process and extract such relationships. Such data can be of significant value to many fields of science, including the field of law. Performing inference and argumentation in a proper and automatic manner can be used for many purposes and can assist those using legal texts in their daily work.

A. Causality relationship

We can define causality as a relationship between events e_1 and e_2, such that the occurrence of event e_1 results in the occurrence of event e_2 [1]. The following division of causality is made [2]:

- Explicit causality, which occurs in a sentence in the form of overt, often with conjunctions or causal phrases, such as in the sentence: “I did not attend the event because I was not invited.”
- Implicit causality, which does not occur in overt form overt, and can often be split into several sentences, such as in the sentences: “Drive slower. It’s slippery.”

It should be noted that in some cases the sentences, causes or effects may be unequal to each other, such as “The reason for the verdict was the evidence supporting the defendant’s guilt, but also the lack of cooperation on his part”. In this example, both “evidence supporting the defendant’s guilt” and “a lack of cooperation on his part” are causes in a sentence. Cause and effect can also be nested as in the sentence: “Refusal to testify or failure to appear at trial cause the court’s disfavor and the defense counsel’s concern.”. In this case, both “refusal to testify” and “failure to appear at trial” may cause further effects. Associations may also share certain parts with each other. In the next example, the effect of the first cause is also a cause for the next effect: “The defendant’s inappropriate behavior caused agitation in the courtroom, as a result of which the court had to cancel the hearing”.

Causality can be single-sentence or multi-sentence. Single-sentence is often combined with so-called overt causal conjunctions and phrases, which we can divide into:

- causal conjunctions: “because”, “as”, “cause”,
- result phrases: “as a result”, “due to”, "because of”,
- conditional phrases: “if ... then ...”.

In the case of implicit or multi-sentence causality, it is up to the reader to use basic knowledge to analyze and infer to detect it. These are much more complicated and therefore more difficult to analyze [3].

B. Practical uses

Detecting causal relationships in texts is of immense value and can be used in predictive and analytical tasks [3]. Having such information can be helpful in many fields [4], such as

- medicine, when analyzing medical cases,
- learning about the causes of security incidents,
- learning about the effects of natural disasters, etc.

In the context of the legal field, information about such relationships can carry a lot of value, for example, in the context of adjudicating court cases (especially in countries where the law of precedent is used, such as the US) and can be an important aid to judges in formulating a verdict. Also for prosecutors, or attorneys, such information can help in taking the right strategy in the courtroom.

II. Research status

Two main types of methods can be found in articles and scientific papers to detect cause-and-effect relationships [3]:

- methods based on patterns or rules [5], [6], [7],

- methods based on machine learning techniques [8], [17], [18], which we can divide into statistical methods (e.g. using decision trees, Naive Bayes algorithm or linear regression) and deep learning methods (neural networks).

The first type manifests weakness in many areas due to the need to create very sophisticated rules or patterns, thus requiring a lot of domain knowledge. Methods based on machine learning, on the other hand, although built without human intervention, need to be programmed and trained, thus requiring a lot of hardware and time resources.

A common approach appearing in the literature, is the use of a two-step causality extraction: first the detection of candidates is done, and then the classification of relationships. In this approach, there can be so-called cascading errors [19], i.e. errors that, when present in the first step, can significantly affect the results of the next step.

A. Data preparation

In order for the model to be trained, proper data preparation is required. The authors of [13] used the technique of labeling words using the Cartesian product of entity and relation tags, and then assigned a unique tag to the word. On the other hand, in [8] a new approach was proposed, the so-called "BIO and CEEmb" labeling of words based on tags: cause (C), effect (E) and embedded causality (Emb). An additional step is to mark each word as the beginning of the cause/effect (B), the continuation of the cause/effect (I), and another word (O). This approach makes it possible to formulate causal triples. Suppose we have a sentence, "The court refused to continue the trial due to the absence of the defendant." After analyzing this example, we can formulate a causal triple, where two events are divided by the type of relationship (in this case, cause-effect): "refusal to continue the trial, cause-effect, the absence of the defendant".

However, some tagging schemes, such as the one proposed in [17], cannot identify overlapping relationships. To solve this in [8], the authors use the "Tag2Triplet" algorithm, which allows the extraction of nested relationships in which individuals can be part of multiple ones. For example, the sentence "As a result of the incident, the plaintiff was unable to testify, leading to incorrect conclusions." contains an effect, i.e. the lack of testimony, which is also the cause of another effect, i.e. "incorrect conclusions."

B. Detecting and extraction

Following the determination process, the dominant approach is using recurrent LSTM neural networks in varieties with connection to conditional random fields [8] or in the Bi-LSTM type [4]. Some works in [21] or [22] have focused on detecting causality per se without dividing it into full relations (they detect sentences in which such a relation exists without dividing them into cause and effect), and some, e.g. in [4] or [23] focus on identifying linguistic expressions useful in describing causality (such as conjunctions and causal phrases).

In [8], [24], authors also point out that the use of word embedding layers makes a significant contribution to improving the performance and overall results of causality extraction. To improve the detection of relationships that remain far apart, various techniques are being introduced, such as the so-called self-suggestion mechanism [25], which, unlike the classical LSTM approach, can lead to a connection between arbitrarily distant words [26], and thus detect relationships between words in a more sophisticated way. This is because the meaning of a word is defined in the context of its entire surroundings, and not just (as in simple recurrent networks) based on what is immediately before or after it. The main problem in causality extraction is the embedding of such relationships in the text. On the other hand, extracting prepositions or effects without traditional conjunctions ("because", "since", "if", etc.) is an extremely difficult task [8].

III. Experiment

Causality can often be buried very deeply in a text, and even a person himself may have trouble pointing it out. Extracting such relationships from legal texts significantly narrows the corpus of words that can be used. In addition, the collection can be narrowed even further when focusing on a specific type of legal texts, such as the texts of court judgments. Among current studies, such experiments, i.e. causality analyses for legal texts, are lacking, especially when talking about Polish.

In the experiment, we focused on the extraction of explicit causality at the sentence level. This task is divided into two parts — the first goal is to indicate whether a sentence contains a causal relationship, while the second is to label and extract parts of such relationships. Not only semantic analysis becomes important here, but also the construction of the sentence itself. The biggest problem in this type of experiment is undoubtedly the lack of a suitable learning set in Polish. Therefore, it became necessary to manually prepare such a set before starting further analysis. In order to perform it, recurrent neural networks with LSTM-type cells were used, along with layers of word embeddings.

A. Data preparation

To conduct the experiment, it was necessary to prepare a dataset. For this purpose, legal texts were used, specifically court judgments from several open sources [9], [10], [11], [12]. The total number of judgment texts was 150. Using the author's script (adopting the beautifulsoup library in Python [13]), court judgments were downloaded from the above-mentioned sources in HTML format, and then converted to text and divided into sentences (using the NLTK library [14]). Each document has been marked accordingly (as indicated below). Two datasets were manually prepared for the experiment: the first set in order to perform binary classification on it — each sentence was assigned a positive or negative label, depending on the presence or absence of a cause-and-effect relationship in it. The second set was prepared based on sequence labeling, where each word in a sentence was assigned a label indicating its type in a sentence with causality. Both collections were prepared manually, requiring human intervention. We marked

Example 3.1 (Examples of elements of the first set):

```
Natomiast przedawnieniu podlega samo
ustalenie odszkodowania, gdyż wg woli
ustawodawcy następuje ono w formie
decyzji administracyjnej.;1

Niezbędne jest dodatkowo wykazanie
konieczności wyjaśnienia zakresu
sprawy.;0
```

TABLE I
THE NUMBER OF CLASSES IN THE FIRST SET

	Number of elements	Percentage
Class 0 (no relation)	22060	92.01%
Class 1 (with a relationship)	1914	7.99%
Total	23974	100%

the data manually on our own, then we verified it (for this purpose we used Doccano software [15]).

1) First dataset: In the first set, each sentence was labeled, i.e. assigned a corresponding class, according to the presence of a causal relationship (class 1, positive) or its absence (class 0, negative), as shown in the Example 3.1. It is worth noting that this set is not a balanced set — the negative clause significantly dominates (Table I), which has implications for further text analysis.

2) Second dataset: Each sentence that contained a cause-effect relationship was additionally labeled, i.e. each word was given a membership in one of the groups: cause (class 0), effect (class 1), causal phrase (class 2), other (class 3). A collection of such sentences, divided into words, has been marked accordingly (Example 3.2). Each element was labeled in such a way that it could contain multiple consecutive words within it (Table II). The tagging method is a modified version of the method presented in the [20].

Example 3.2 (Example element of the second set):

```
w ocenie sądu okręgowego nagrody z
zakładowego funduszu nagród wypłacone
wnioskodawcy niepodlegają uwzględnieniu
przy ustalaniu podstawy wymiaru renty
gdyż nie były zaliczane do wynagrodzeń
osobowych 11111111111111111111112000000
cause-effect-sentence
```

TABLE II
THE NUMBER OF CLASSES IN THE SECOND SET

	Number of elements	Percentage
Class 0 (cause)	1748	32.43%
Class 1 (effect)	1729	32.08%
Class 2 (causal phrase)	1774	32.91%
Class 3 (other)	139	2.58%
Total	5390	100%

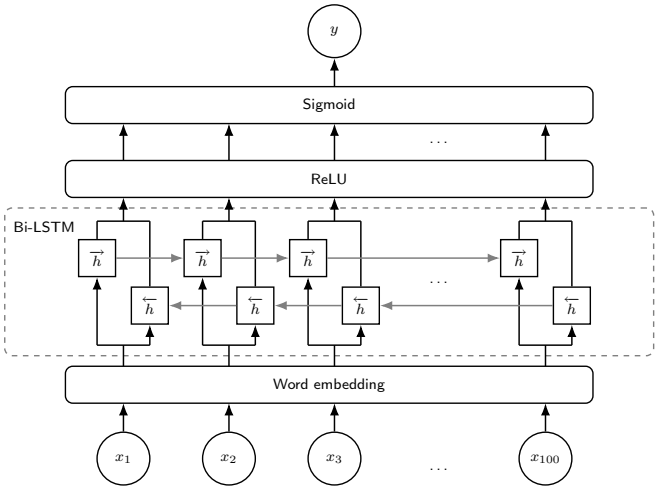

Fig. 1. The architecture of the network responsible for marking a sentence as having causality or not

B. RNNs structures

The program for detecting explicit cause-effect relationships was built on the basis of two separated neural networks. For this purpose, the Tensorflow library and the Keras interface were used [16]. The first network is responsible for binary classification of whether a cause-effect relationship is present in a sentence. The second network is tasked with performing cause-and-effect extraction, i.e. labeling a sentence with a cause-and-effect relation, assigning each word a token of the appropriate class. In both cases, validation of the correctness of the trained models is carried out at the end of the subroutines.

1) First network: The input data is properly prepared before entering the network, by dividing it into tokens, removing punctuation and whitespace characters. The set is divided in a 7:3 ratio into a learning set and a validation set. The next step is to transform the sentences into a dense feature vector using a set of word embeddings for the Polish language [27], [28], i.e. a 100-dimensional corpus containing all parts of speech, created using the CBOW architecture. Based on the subset counts, the weights of each class are calculated (due to the unbalanced dataset). The data then becomes the input for a recurrent neural network in the Bi-LSTM variant, in which the first layer is the word embedding layer (loaded earlier). The detailed architecture of the network is shown on Fig. 1.

The model was created using standard binary cross entropy as a loss function and the adam optimization algorithm. After training, the model is tested with a validation set and evaluated (precision, recall, F1 and accuracy values are calculated, as well as the ROC curve and the value under the AUC curve). The training process took place in 8 epochs, during which all the above metrics were measured.

2) Second network: The task of the second neural network is to extract cause-and-effect relationships from a sentence evaluated positively as containing causality. Each word must be assigned one of four classes: cause, effect, connective phrase or another word. The input sentences, as in the case

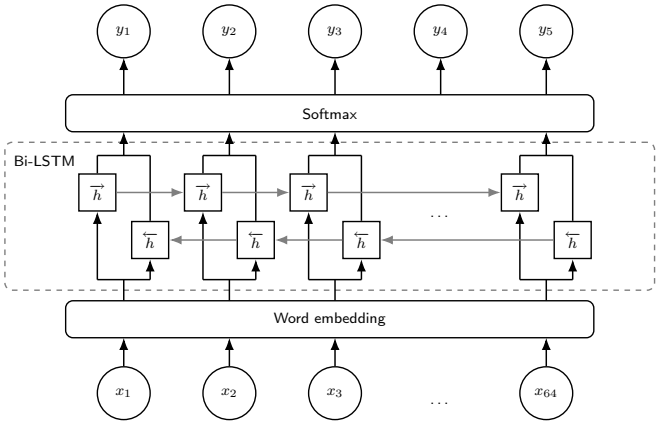

Fig. 2. The architecture of the network responsible for extracting the cause and effect parts

TABLE III
THE RESULTS OF THE FIRST NEURAL NETWORK

	Precision	**Recall**	**F1**
Class 0	0.81	0.89	0.85
Class 1	0.99	0.98	0.99
Macro	0.90	0.94	0.92
Weighted	0.98	0.98	0.98

of the first neural network, undergo preprocessing (identical to that described above), and then go as input to the neural network in the Bi-LSTM variant (Fig. 2). Due to the small size of the collection, cross-validation was used in the validation process, i.e. the collection was divided into ten parts and trained nine of them at a time, and tested the last one. After the training process, the results of the network are validated using the metrics of precision, accuracy, recall, and F1 index, both for each class and the entire collection. The training process took place over 10 epochs, during which all of the above metrics were measured.

IV. RESULTS

The following tables present the values of the metrics for each set and each program. Table III shows the validation results of the first neural network tasked with binary classification. The accuracy for the entire set was 97.68%.

The value of AUC = 0.9822, which shows that the classifier can correctly distinguish class elements. The high precision is maintained throughout the learning period of the model due to issues related to the unbalanced dataset, described below. Details of the values of the metrics at training time (at a specific epoch) are shown in Fig. 3-7. Noteworthy, this curve gives an incomplete picture of the classifier, due to the unbalanced dataset. The more important information is the values for the class with causality sewn in, the results of which no longer look so good (as can be seen in the confusion matrix of validation set in Table IV).

As the results indicate, the classification of such relationships is not a simple task, but to some extent it is feasible.

TABLE IV
CONFUSION MATRIX OF THE FIRST NEURAL NETWORK

	Actually positive	**Actually negative**
Predicted positive	477	58
Predicted negative	109	6549

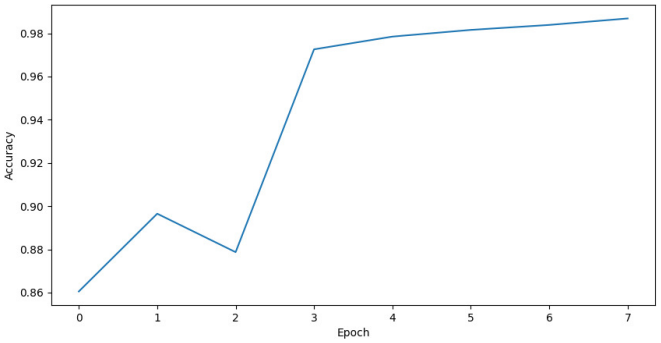

Fig. 3. Accuracy in training the first neural network

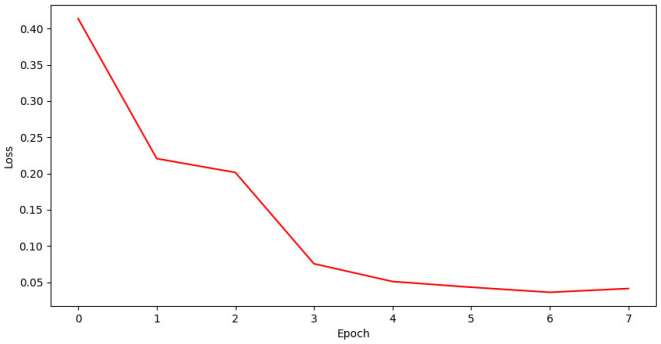

Fig. 4. Loss in training the first neural network

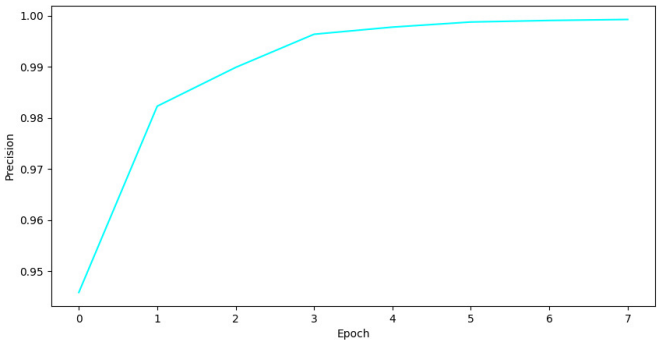

Fig. 5. Precision in training the first neural network

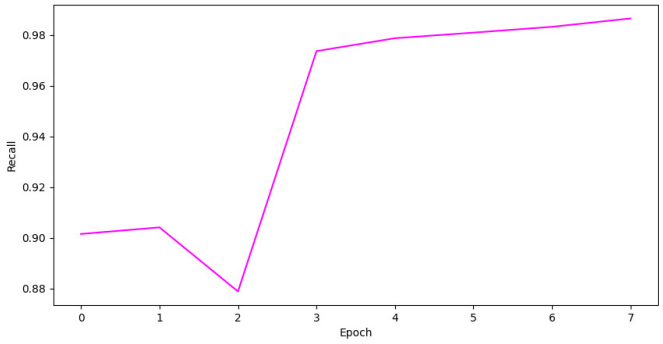

Fig. 6. Recall in training the first neural network

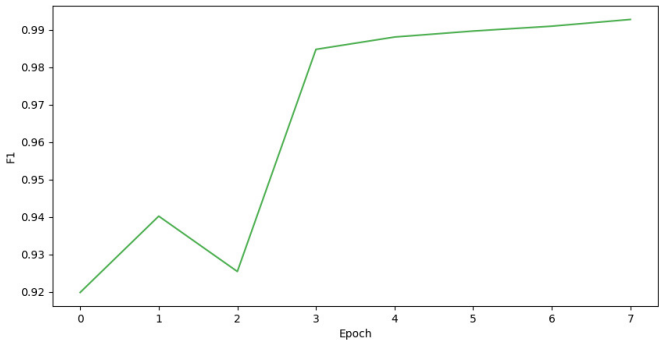

Fig. 7. F1 process of training the first neural network

This is greatly influenced by words that are parts of causal phrases, but it should be noted that there is never such certainty. For example, when a sentence contains the word "albowiem", which often occurs in legal language, this may or may not indicate conditionality. The word "bowiem" in most cases separates the cause and effect parts, but there are also exceptions to this.

Table V shows the results for the second neural network, which was tasked with extracting the parts belonging to causal relationships. The results here are much worse. What stands out here is the better result of the causal phrase class, due to the frequent occurrence of the same phrases and words. The results here are probably also influenced by the small collection. In contrast, class with other words (class 3) performs in a clearly negative way, given the problem of indicating it in a sentence, because there are no special rules formulated in the experiment for the occurrence of such a class.

TABLE V
THE RESULTS OF THE SECOND NEURAL NETWORK

	Precision	**Recall**	**F1**
Class 0	0.58	0.58	0.58
Class 1	0.50	0.51	0.51
Class 2	0.72	0.89	0.80
Class 3	0.18	0.06	0.09
Macro	0.60	0.61	0.60
Weighted	0.83	0.83	0.83

It should also be noted that legal texts (especially court judgments) often have sentences that are very rote in their construction, i.e. contain many subordinate sentences, which also affects such analysis. The network also did not cope when a word or phrase indicating causality was located at the beginning of a sentence. In such a case, the word "ponieważ" does not separate the causal part from the effect, so the network's results were subject to high error.

V. CONCLUSIONS

The biggest problem with the argument extraction experiment became the lack of a suitable training set. There is no such set for the Polish language in the sources, which made it necessary to create such a set manually. This was a tedious activity, but at the same time required adequate attention. Closing the corpus of words to the texts of court rulings, significantly simplified the analysis and marking of sentences, due to the orderly structure of the text, often containing similar causal phrases. Judgment texts, like other legal texts, are often written in correct language, but stylistic, punctuation and even spelling errors can be found among them (unlike, for example, the texts of statutes). The structure of a court decision itself looks very similar, regardless of the court or its type (division into a operative part, justification or cited provisions).

In the case of the first set (the input for the first neural network tasked with binary classification), sentences that have a causal relationship in them make up a small percentage of the set. Hence, it is necessary to set up the neural network in such a way as to notify it of the greater importance of certain elements of the set. The reason for using such a set is to reflect the real ratio of sentences that contain a causal relationship to those that do not. The use of a word embedding layer with a trained set of vectors for the Polish language also has a broad impact on better results.

In some cases, the word occurs with cause (without effect), indicating that causation is missing at the sentence level. Thus, it cannot be assumed that syntactic analysis alone would carry significant information about the semantics of the sentence, but it would be largely sufficient. Reviewing the results, we can note the following regularities. For example, a sentence containing a causal connective phrase has a high degree of certainty about the occurrence of a cause in it. On the other hand, a sentence that does not have such a phrase with the highest probability is assigned to a class with no such relationship.

VI. FUTURE WORKS

To develop the topic of causal relationship extraction in the future, it would therefore be important to create a suitably large and diverse test dataset. Semantic analysis at the level of the whole document, and not just at the sentence level, would also be an important element. This would allow detection of arguments that are implicit relationships (sewn into the text), often found in different parts of the document. As research in the field of detecting such relationships shows,

this task is not easy. When analyzing texts in Polish, we often also have to pay attention to other elements absent in other languages, which makes such texts significantly more difficult to analyze semantically for causality. The resulting data from this experiment can successfully serve for further research and be the basis for other tasks in the area of machine learning in the field of law.

ACKNOWLEDGMENT

Special thanks to dr Tomasz Żurek for inspiration and his help with defining the described problem and dr Andrzej Bobyk for providing his knowledge and experience.

REFERENCES

[1] Stanford Encyclopedia of Philosophy, *Causal Models*, 2022 https://plato.stanford.edu/entries/causal-models/

[2] E. Blanco, N. Castell, and D. Moldovan, *Causal relation extraction,* Proceedings of the Sixth International Conference on Language Resources and Evaluation, 2008, pp. 310

[3] Yang J, Han S. C. and Poon J. *A survey on extraction of causal relations from natural language text*, Knowledge and Information Systems 64, 2022, pp. 1161-1186, https://doi.org/10.48550/arXiv.2101.06426

[4] T. Dasgupta, R. Saha, L. Dey, and A. Naskar, *Automatic Extraction of Causal Relations from Text using Linguistically Informed Deep Neural Networks*, Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue, Melbourne, Australia. Association for Computational Linguistics, 2018, pp. 306-316, http://dx.doi.org/10.18653/v1/W18-5035

[5] C. S. G. Khoo, J. Kornfilt, R. N. Oddy, and S. H. Myaeng, *Automatic Extraction of Cause-Effect Information from Newspaper Text Without Knowledge-based Inferencing,* Literary and Linguistic Computing, Volume 13, Issue 4, 1998, pp. 177–186, https://doi.org/10.1093/llc/13.4.177

[6] C. S. G. Khoo, S. Chan, and Y Niu, *Extracting Causal Knowledge from a Medical Database Using Graphical Patterns*, Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics, Hong Kong, Association for Computational Linguistics, 2000, pp. 336–343, http://dx.doi.org/10.3115/1075218.1075261

[7] R. Girju, D. Moldovan, *Text Mining for Causal Relations*, Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference, 2002, pp. 360–364

[8] R. Girju, *Automatic Detection of Causal Relations for Question Answering*, Proceedings of the ACL 2003 workshop on Multilingual summarization and question answering, Association for Computational Linguistics, 203, pp. 76–83, http://dx.doi.org/10.3115/1119312.1119322

[9] Portal Orzeczeń Sądów Powszechnych, https://orzeczenia.ms.gov.pl/

[10] Wyrok.org — Największa baza wyroków w Polsce, https://wyrok.org/

[11] Centralna Baza Orzeczeń Sądów Administracyjnych, https://orzeczenia.nsa.gov.pl/

[12] Dziennik wyroków i ogłoszeń sądowych, https://www.ebos.pl/

[13] Beautiful Soup Python library, https://www.crummy.com/software/BeautifulSoup/bs4/doc/

[14] NLTK: Natural Language Toolkit, https://www.nltk.org

[15] H. Nakayama, T. Kubo, J. Kamura, Y. Taniguchi and X. Liang, *Doccano: Text Annotation Tool for Human*, 2018, https://github.com/doccano/doccano

[16] M. Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*, 2015, https://doi.org/10.48550/arXiv.1603.04467

[17] A. Sorgente, G Vettigli, and F. Mele: *Automatic extraction of cause-effect relations in Natural Language Text*, Proceedings of the 7th International Workshop on Information Filtering and Retrieval co-located with the 13th Conference of the Italian Association for Artificial Intelligence, 2013, pp. 37–48

[18] S. Zhao, T. Liu, S. Zhao, Y. Chen, and J. Nie, *Event causality extraction based on connectives analysis*, Neurocomputing 173, 2016, pp. 1943–1950, https://doi.org/10.1016/j.neucom.2015.09.066

[19] Z. Li, Q. Li, X. Zou, and J. Ren, *Causality Extraction based on Self-Attentive BiLSTM-CRF with Transferred Embeddings*, Neurocomputing 423, 2021, pp. 209, https://arxiv.org/abs/1904.07629

[20] S. Zheng, F. Wang, H. Bao, Y. Hao, P. Zhou, and B. Xu, *Joint Extraction of Entities and Relations Based on a Novel Tagging Scheme*, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2017, pp. 1227–1236, http://dx.doi.org/10.18653/v1/P17-1113

[21] T. N. de Silva, X. Zhibo, Z. Rui, and M. Kezhi, *Causal Relation Identification Using Convolutional Neural Networks and Knowledge Based Features*, International Journal of Computer, Electrical, Automation, Control and Information Engineering 11 (6), 2017, pp. 697–702, https://doi.org/10.5281/zenodo.1130679

[22] C. Kruengkrai, K. Torisawa, C. Hashimoto, J. Kloetzer, J. H. Oh, and M. Tanaka, *Improving Event Causality Recognition with Multiple Background Knowledge Sources Using Multi-Column Convolutional Neural Networks*, Proceedings of the AAAI Conference on Artificial Intelligence 31(1), 2017, https://doi.org/10.1609/aaai.v31i1.11005

[23] J. Dunietz, J. Carbonell, and L. Levin, *DeepCx: A transition-based approach for shallow semantic parsing with complex constructional triggers*, Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 1691–1701, http://dx.doi.org/10.18653/v1/D18-1196

[24] A. Akbik, D. Blythe, and R. Vollgraf, *Contextual String Embeddings for Sequence Labeling*, Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 1638–1649

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, *Attention is All you Need,* Advances in Neural Information Processing Systems 30, 2017, pp. 6000–6010, https://doi.org/10.48550/arXiv.1706.03762

[26] Z. Tan, M. Wang, J. Xie, Y. Chen, and X. Shi: *Deep Semantic Role Labeling with Self-Attention*, Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, 2018, https://doi.org/10.48550/arXiv.1712.01586

[27] A. Przepiórkowski, M. Bańko, R. L. Górski, and B. Lewandowska-Tomaszczyk, *Narodowy Korpus Języka Polskiego*, Wydawnictwo Naukowe PWN, Warsaw, 2012

[28] A. Mykowiecka, M. Marciniak, and P. Rychlik, *Testing word embeddings for Polish*, 2017, http://dsmodels.nlp.ipipan.waw.pl