BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **18**, No 1
Sofia • 2018 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2018-0013

Graph-Based Complex Representation in Inter-Sentence Relation Recognition in Polish Texts

Arkadiusz Janz, Paweł Kędzia, Maciej Piasecki

Wrocław University of Science and Technology, 50-370 Wrocław, Poland
E-mails: *arkadiusz.janz@pwr.edu.pl pawel.kedzia@pwr.edu.pl maciej.piasecki@pwr.edu.pl*

***Abstract*:** *This paper presents a supervised approach to the recognition of Cross-document Structure Theory* (*CST*) *relations in Polish texts. Its core is a graph-based representation constructed for sentences. Graphs are built on the basis of lexicalised syntactic-semantic relations extracted from text. Similarity between sentences is calculated as similarity between their graphs, and the values are used as features to train the classifiers. Several different configurations of graphs, as well as graph similarity methods were analysed for this task. The approach was evaluated on a large open corpus annotated manually with 17 types of selected CST relations. The configuration of experiments was similar to those known from SEMEVAL and we obtained very promising results.*

***Keywords*:** *Cross-document structure theory, CST, supervised learning, graph-based representation, logistic model tree, LMT, support vector machine, SVM.*

1. Introduction

Due to the rapid expansion of sources of electronic text, the volume of data that we deal with has increased significantly. Among large volumes of data available one can find a lot of redundant information, e.g., supplementing, overlapping etc. Manual aggregating and synthesizing valuable information from a massive input is laborious and difficult. However only a fraction of the input is the core or the most relevant. The aim of multi-document discourse parsing is to discover the relations or dependencies linking text passages from different documents. The relations we are aiming for are not limited only to the relations between event descriptions. Recognition of discourse relationships linking texts can be useful in many information retrieval applications, and may help to deal with information extraction and information management.

The Cross-document Structure Theory (CST) (R a d e v, [27]) introduces an organized structure of semantic links connecting topically related texts. CST relations, if only recognised correctly for text fragments, can provide a map of the

document semantic structure and, e.g., may have a positive impact on effectiveness in supporting multi-document summarisation, e.g., K u m a r et al. [14].

CST relations can be categorized according to some specific criteria, taking into account the form of related texts, as well as the overlap of their information content. An example of relation referring to the degree of information overlap may be *Subsumption*, where some information is fully repeated in both sentences, but there is also some additional information included only in one of these sentences, thus IC(S_1)\IC(S_2) = Ø, but not vice versa:

S_1: *Nie ma jeszcze żadnych informacji na temat ofiar.*

(*Information on the victims remains unknown.*)

S_2: *Żadne dalsze szczegóły nie są jeszcze znane.*

(*Further details about the incident remain unknown.*)

Overlap relation means a partial intersection of information content where $IC(S_1)\backslash IC(S_2) = X$, $IC(S_2)\backslash IC(S_1) = Y$ and $IC(S_1)\backslash\cap IC(S_2) = Z \neq \emptyset$, Identity and Paraphrase relations are representing full equivalence of information $IC(S_1) = IC(S_2)$, but for *Paraphrase* this information is expressed using slightly different forms of words and expressions. CST introduces even more relations taking into account the nature of certain temporal dependencies that may occur between a pair of texts referring to the same event, e.g., *Historical Background*, where first sentence describes a historical context of the information or event occurring in the second sentence, or *Fulfilment* where the first sentence confirms the occurrence (fulfilment) of an event that was announced in the second sentence.

However, due to the large number of relations and often subtle differences between them, CST relation recognition is known to be much harder than Textual Entailment (TE) recognition. TE depends on a binary decision whether one piece of text semantically entails another one due to their content, while CST is a model of more general use, but more difficult to achieve good results, especially when a classifier is trained on a domain different than the domain of its application.

Differences in the definitions of *Description*, *Follow-up* or *Elaboration* indicate some potential difficulties that may arise when we want to recognize certain types of relations. In the case of *Description*, the new additional information is about the current, non-historical nature of an event, e.g., the first sentence describes an object or entity appearing in the second sentence. *Elaboration* provides some additional details regarding the event, but generally the sentences convey the same core information. *Follow-up* provides some unrevealed facts about the event but appearing after occurrence of this event, thus it may be some kind of description for related events.

Our goal is to build a tool for the recognition of CST relations in Polish texts. Firstly, we limited the problem to recognition of relations between sentence pairs that is even a harder task because of the limited text material which is to be processed. For training we used a part of the KPWr Corpus (B r o d a et al. [2]) based on Polish Wikinews (**https://pl.wikinews.org**). In the work presented here, we focus on the 17 relations with the largest coverage in the corpus and which seem to be the most important from the point of view of applications.

2. Related works

In Zhang, Otterbacher and Radev [39] CST relations were recognized by a supervised approach with boosting on the basis of simple, lexical, syntactic and semantic features, extracted from sentence pairs. The evaluation was performed in two steps: binary classification for relationship detection, and multi-class classification for relationship recognition. The same set of features was used in both steps. This idea was expanded in Zhang and Radev [38] by leveraging both labelled and unlabelled data. The exploitation of unlabelled instances improved the performance. Boosting technique was used in combination with the same set of features to classify the data in CSTBank (Radev, Otterbacher and Zhang [28]). Relation detection was significantly improved to F-score = 0.8839. However, recognition of the relation type was still unsatisfactory, at least from the point of view of potential applications.

Aleixo and Pardo [1] is one of a few works that address recognition of CST relations for languages other than English. They utilised CST in search for topically related Portuguese documents. They applied a supervised approach based on similarity measures calculated for sentence pairs from different documents: Zahri and Fukumoto [37] applied the supervised learning to identify a limited set of CST relations*: Identity, Paraphrase, Subsumption, Elaboration and Partial Overlap*. They were used in the multi-document summarization task. SVM algorithm was used and examples from CSTBank. The features of Aleixo and Pardo [1] were expanded with:

1) cosine similarity of word vectors,

2) intersection of common words measured with the Jaccard Index,

3) an indicator of longer sentence (1 if S_1 was longer, 0 if equal, –1 if S_1 was shorter),

4) and uni-directional word coverage ratio ($S_1 \rightarrow S_2$ and $S_2 \rightarrow S_1$).

Kumar et al. [12] followed Zahri and Fukumoto [37], but restricted the set of relations further down to four: *Identity, Subsumption*, *Overlap and Elaboration*.

Four features were used:

1) tf-idf based cosine sentence similarity,

2) words coverage ratio,

3) sentence length difference,

4) and the indicator of longer sentence.

The best performance of SVM in relation recognition was: for *Identity* F=0.91, Subsumption 0.59, *Elaboration* 0.54, and 0.62 for *Overlap*. For the same relations Kumar, Salim and Raza [13] presented results obtained with SVM, a Feed-Forward neural network and CBR (Case-based Reasoning). The features of Zahri and Fukumoto [37] were extended with the Jaccard based similarity of noun phrases and verb phrases from the compared sentences. The best result was achieved with CBR based on the cosine similarity measure. It expressed improved results than in Kumar et al. [12]: *Identity* 0.966, *Subsumption* 0.803, *Description* 0.786, and 0.722 for *Overlap*.

Due to the ambiguity in the interpretation of certain CST relationships Maziero, Jorge and Pardo [22] proposed several refinements to CST in order to reduce the ambiguity. They improved definitions by introducing several additional constraints on the co-occurrence of different relations in texts. The CST taxonomy was amended by adding a division based on the form and information content of relations. The improved model was used in evaluation of supervised CST relation recognition in three different settings: *binary* (a separate classifier per each relation), *multi-class* and *hierarchical* (facilitating the proposed taxonomy of relations). The applied features included: sentence length difference, ratio of shared words, sentence position in text, differences in word numbers across PoSs, and the number of shared synonyms between sentences. SVM, Naive Bayes and J48 decision tree were used for classification. The J48-based classifier achieved the best score in all three approaches. The average F-measure for multi-class scheme was 0.403, while for the binary scheme: 0.673 (but without calculation of the final decision) and for the hierarchical: 0.724.

3. Dataset

For the development and evaluation of the proposed approach, we utilised a dataset of sentence pairs annotated with CST relations from the KPWr Corpus Broda et al. [2], see Section 1. All annotated sentences come from a corpus collected from Wikinews (**https://pl.wikinews.org**) materials written in Polish. The corpus contains 11,949 complete documents that were clustered on the basis of their cosine similarity and split into groups of 3 news each. More specifically, for every document in the corpus the list of 20 most similar documents was created, but only 3 with the highest similarity formed a group. These groups include the most similar, potentially topically related documents. Afterwards, we prepared a set of bundles for manual annotation process – every with 10 triples $\{D_1, D_2, D_3\}$ of most similar documents – and we randomly assigned them to the annotators. Finally, 96 bundles covering more than 2,800 documents were analysed in order to discover new instances of CST relations. The imposed similarity structure facilitated searching for sentence pairs linked by a CST relation. We collected manually annotated pairs of sentences representing new instances of CST relations to create the Gold Reference Subcorpus, but for the final corpus WUT CST (**https://clarin-pl.eu/dspace/handle/11321/305**) we have rejected uncertain CST instances with inconsistent annotations. This means that our WUT CST corpus contains only CST instances with almost homogenous annotations assigned by at least $n - 1$, $n > 2$ annotators. The final distribution of collected CST instances in our WUT CST corpus is presented in Fig. 1.

A corpus, with similar distribution of discourse relations linking multiple documents, was also introduced in Cardoso et al. [6]. It was built from texts published in journals in Brazilian Portuguese.

Selected sentences from our corpus were manually annotated with CST relations by at least by 3 annotators (linguists) each.

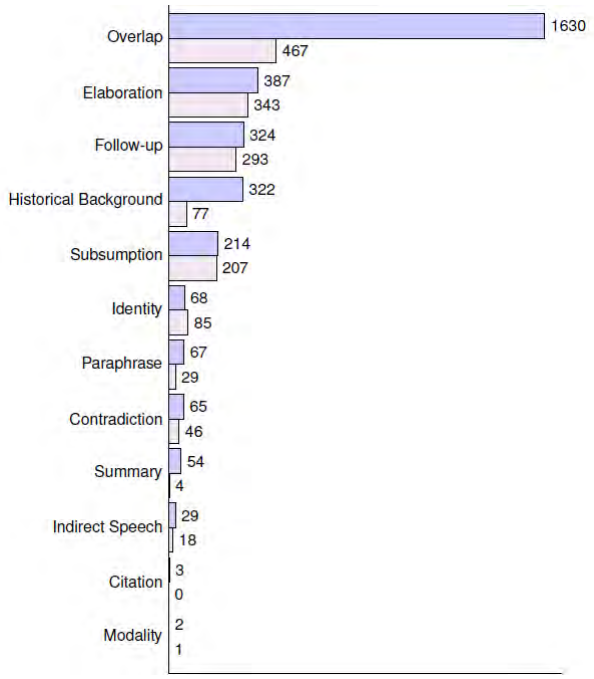

Fig. 1. Distribution of relations in the Gold Reference Subcorpus (GRS) compared to CSTNews. (from C a r d o s o et al. [6])

Each annotator was exploring the corpus independently, in order to find and annotate inter-document relations inside document groups.

The annotators followed the guidelines used for the construction of CSTBank (R a d e v, O t t e r b a c h e r and Z h a n g [28]) that were slightly adapted to Polish.

4. Features for classification

4.1. Baseline features

As a starting point for the selection of features describing sentence pairs we used the set features proposed by M a z i e r o, J o r g e and P a r d o [22]. Our set includes commonly-used, lexical, syntactic and semantic features that were applied for the detection and recognition of CST relationships in supervised approaches. They focus on the grammatical forms and properties of the linked sentences:

Shared lemmas – the number of lemmas (A lemma is a basic morphological form that represents a set of inflected word forms that differ only in the value of grammatical categories) shared by two sentences,

Shared PNs –the number of Proper Names (automatically detected) shared by two sentences,

Longest Common Substring – the length of the longest common continuous sub-string of word forms from the two sentences,

Longest Common Subsequence – the length of the longest common sub-sequence, but the sequences can be discontinuous, i.e., the found sequence can be separated by some other tokens in one or both sentences,

Cosine similarity – the cosine similarity of vectors of the frequency of lemmas, built for both sentences, on the basis of the frequency of lemmas (Bag-of-Words model) in each of the two sentences,

Is Longer – equals 1 if the first sentence is longer, 0 for equal, –1 if the second is longer,

Shared synsets – the number of synsets shared by the two sentences which is normalized by the number of all synsets in the shorter sentence. This normalization takes into account the fact, that the number of common meanings will be less than or equal to the length of the shorter sentence (to make the feature insensitive to sentence length differences),

PoS similarity – cosine measure of vectors of the frequencies of different Part of Speech in both sentences – the grammatical classes from the tagset of the Polish National Corpus and the feature can be parametrised by the expected granularity of classes, e.g., mapped to the traditional Parts of Speech. In our research, this granularity was set only to 4 basic PoS.

SVO Index – the Jaccard Index calculated for vectors of frequencies of triples: subject, verb, and object for both texts.

The above features were used as a baseline model for the description of text pairs, and compared later with the graph-based representation proposed in the following subsections. Several language tools were used to enrich texts for feature extraction. These tools were selected to annotate documents at different levels of text analysis, i.e., Morfeusz (Woliński [34]) – a morphological analysis, WCRFT (Radziszewski [29]) – tagger, Liner2 (Marcińczuk, Kocoń and Janicki [20]) – recognition of Proper Names, Maltparser (Nivre et al. [23]) adapted to Polish (Wróblewska, [36]), – dependency parsing, WCCL (Radziszewski, Wardyński and Śniatowski [31]) – recognition of multi-word expressions from plWordNet (Maziarz et al., [21], Piasecki, Szpakowicz and Broda [25]), WoSeDon (Kędzia, Piasecki and Orlińska [17], Piasecki, Kędzia and Orlińska [26]) – Word Sense Disambiguation, IOBBER (Radziszewski and Pawlaczek, [30]) – a syntactic chunker, Fextor (Broda et al., [3]) – a tool for feature extraction.

Features that incorporate word semantics need to access additional resources like plWordNet (Piasecki, Szpakowicz and Broda [25]).

The presented set of features encompasses most of the feature sets proposed in literature. So, simple features are not able to discriminate all available relation classes. In most cases they were used only for basic relations that differ in terms of information overlap, e.g., *Identity*, *Equivalence*, *Subsumption* or *Overlap*.

Experiments with the proposed graph-based representation will be next compared with the results of the baseline feature set classifier.

4.2. Graph-based features

The baseline features do not take into account the linguistic structure of sentences being compared. As the parser for Polish has limited accuracy, instead of depending only on the dependency structure produced by the parser we propose a graph-based representation of a sentence (or text) which is flexible and can accommodate results of processing by different language tools.

4.2.1. Graph-based sentence representation

Each sentence S_i is represented as a directed graph G_i. Thus, a relation $R(S_1, S_2)$ between sentences S_1 and S_2 is represented as a relation R between graphs G_1 and G_2: $R(G_2, G_2)$. For them we will calculate a similarity value v_{sim} = SIM(G_i, G_j) where SIM means one of the similarity measures discussed in Section 4.2.2.

Formally, a directed graph $G = (V, E)$ where V is a set of vertices and E is a set of directed and ordered edges e (the maximum number of e can be $V \times V$, in the practise is a subset of $V \times V$). A directed edge $e = (n_s, n_t)$ where n_s is the source node and n_t is the target node, goes from n_s to n_t.

The graphs are built in three steps:

1) creation of the null graphs with nodes of selected types,

2) inserting new edges on the basis of a sentence,

3) and finally (an optional step) merging the graph with subgraphs extracted from external knowledge sources, i.e., plWordNet and SUMO Ontology (Pease [24]).

In **the First Step** an example sentence pair (S_i and S_j) for a relation R is converted into two separate null graphs (A null graph (Harary and Read [8]) is an edgeless graph) G_i and G_j, respectively. Their nodes are of a selected type T (the same for both graphs), represent the words from the sentences and are not connected to each other. If we select more than one node type, we would obtain several null graphs for each sentence. Depending on the chosen node type, one or more words from a sentence S can be represented by the same node. The list of the four possible node types is presented below.

Lemma lower – this is the simplest node type, a node $n_i \in G$ represents a lemma of the word w_i from S, which is converted to lowercase. All words from the sentence with the same lemma (irrespectively of PoS) are represented by the same node in G_j, e.g., for the sentence S_{sample}:

Z ogrodu zoologicznego we Wrocławiu uciekł wąż boa dusiciel i przemieszcza się w stronę Ostrowa Tumskiego;

"From the zoo in Wrocław, a boa constrictor has escaped and is moving towards Ostrów Tumski"

we obtain the following null graph, presented also in Fig. 2.

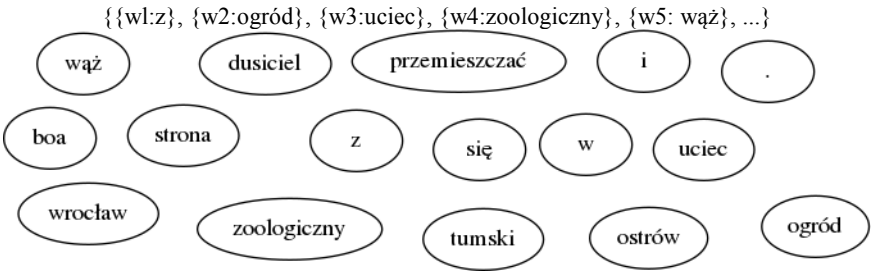

Fig. 2. Null graph built with Lemma lower node type for sentence S_{sample}

Lemma PoS lower – in a similar way to *Lemma lower*, nodes represent lowercased lemmas, but with PoS labels attached to the lemmas, e.g., *cat:n* or the Polish word *piec* can be morphologically disambiguated as a verb or noun *Kasia*

piecze:v ciasto w piecu:n "Kasia is baking a cake in the oven". Using *Lemma lower* type, the words *piecze* "[he/she] bakes" and *piecu* "an oven:inst" will be represented by a single node labelled as *piec*, while in *Lemma PoS lower* type there will be two different nodes: *piec:n* and *piec.v*.

For S_{sample} the node of the type *Lemma PoS lower* are, see also Fig. 3.

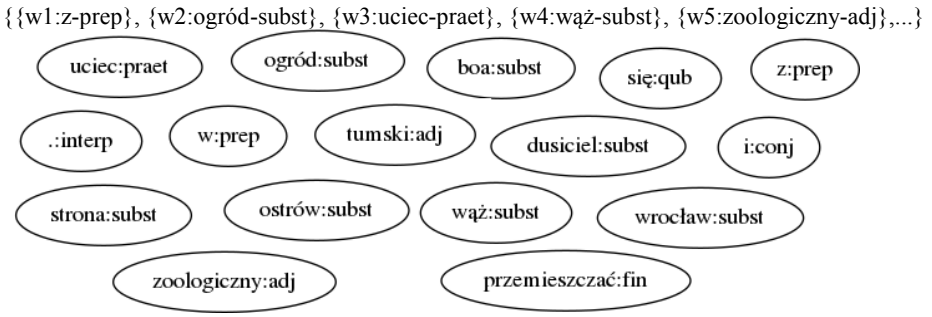

Fig. 3. Null graph built with *Lemma Pos lower* node type for sentence S_{sample}

Synset – nodes represent plWordNet synsets assigned to the words in the sentence as their lexical meanings by WoSeDon. At the beginning, the sentences from the corpus were disambiguated with respect to the word senses by applying WoSeDon (WoSeDon is a tool for recognizing the meanings in Polish texts) tool (Kędzia and Piasecki [15], Kędzia et al., [16], Kędzia, Piasecki and Orlińska [17], Piasecki, Kędzia and Orlińska [26]), each word *w* from sentence *S* has an equivalent in meaning *m*. For S_{sample} and the *Synset* node type, the generated null graph consists of (see also Fig. 4).

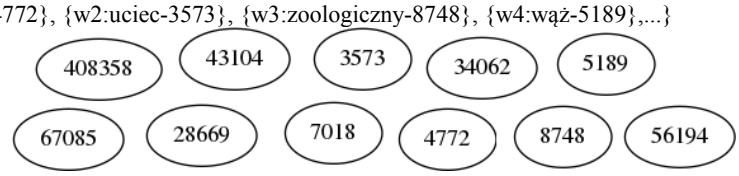

Fig. 4. Null graph built with *Synset* node type for the sentence S_{sample}

Concept – nodes are concepts comming from SUMO Ontology. The concepts are assigned to words in a sentence on the basis of synsets recognised by WoSeDon and the mapping between plWordNet and SUMO (Kędzia and Piasecki [15]). The null graph of *Concept* type for S_{sample} is (Fig. 5).

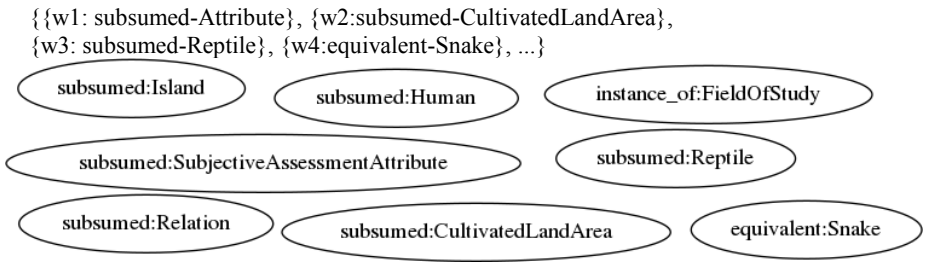

Fig. 5. Null graph built with *Concept* node type for sentence S_{sample}

Second Step – adding the edges.

In **the Second Step** the null graph constructed in the first step is expanded by adding edges between nodes, i.e., e_{type} (n_s, n_t), where n_s, n_t are nodes. If we have multiple null graphs with different node types, we need to expand every null graph from the first step with new edges. The edge types are derived from lexical and semantic relations in a sentence that were automatically recognised with the help of language tools. The e_{type} direction depends on the particular kind of the relation being represented:

w2w – edges represent the word order in a sentence (*word to word*). If a word w_1 occurs in a sentence before word w_2, then there is a directed edge from w_1 to w_2: e_{w2w} : (w_1, w_2).

h2h – head to head represents the relative order of the heads of *agreement phrases* in a sentence. Each sentence is processed first by IOBBER chunker and divided into chunks of three types: Verb Phrase (VP), Noun Phrase (NP) and Adjective Phrase (AdjP), that are next subdivided into smaller, Agreement Phrases (AgP). The relation h2h represents the order of AgPs heads and is added to nodes representing the heads. If a AgP head w_{hi} occurs in a sentence before the AgP head w_{hj}, then the edge is directed from w_{hi} to w_{hj}: e_{h2h}: (w_{hi}, w_{hj}).

ne2ne – an edge type similar to *w2w* and *h2h*, but in which edges represent the order of the named entities *NE* in a sentence. NE is recognized using Liner tool (Marcińczuk [19]). If named entity w_{nei} occurs before w_{nej} in sentence *S*, then a directed edge: e_{ne2ne} : (w_{nei}, w_{nej}), is added to the graph.

malt – edges of this type represent the dependency relations, recognised by the Polish Malt parser (Wróblewska and Woliński [35]). Each dependency relation between two words w_i and w_j, is modelled in the graph as a directed edge with the same direction. If there is a dependency relation dep_{rel} (w_i, w_j), then it is added to the graph as a directed edge with the same direction dep_{rel}:$e_{\text{dep rel}}$ (w_i, w_j). The whole list of names of dependency relations is listed into (Wróblewska and Woliński [35]).

defender – the type similar to the **malt**, but relations come from *Defender* parser which is based on IOBBER chunker and introduces deeper syntactic-semantic relation structures into the representation of NPs, cf (Kędzia and Maziarz [10]). We used both, *malt* and *defender* relations, because in some situations the relations proposed by Malt are incorrect. If there is a dependency for two words w_i and w_j from *Defender*, then it is added as a directed edge to graph: e_{def}(w_i, w_j).

semantic roles – edges marked as *srole* represent semantic roles from *NPSemrel* (The construction of *NPSemrel* is based on hand-written lexicalised syntactic-semantic constraints. They mostly express high precision, i.e., around 60% in the worst cases, but the majority of them is close to 100%. However, the recall is much lower, so F1 measure is typically around 0.5, see Kędzia and Maziarz [10], a Polish shallow semantic parser Kędzia and Maziarz [10]. The dependencies proposed by *Defender* are named with semantic roles, e.g., *agent*, *theme*, see Kędzia and Maziarz [10] for the full set. If semantic role is

assigned to a pair of words: w_i and w_j, a directed edge is added between the nodes representing w_i and w_j: e_{srole}: (w_i, w_j). The edge is labelled with the semantic role.

All types of edges and nodes introduced above were used in our experiments. A single graph G_i represents sentence S_i and contains the edges $E_i \in$ {w2w, h2h, ne2ne, malt, def, srole}.

A simplified graph for sentence S_{example}, with *Lemma PoS* nodes and full set of possible edge types is shown in Fig. 6.

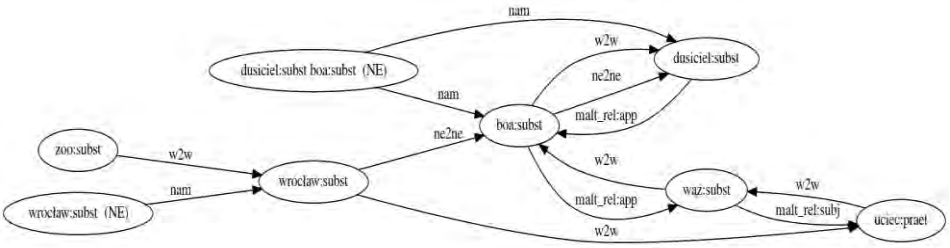

Fig. 6. Graph built for sentence S_{example} with *Lemma PoS* node type and full set of edges types

Another graph for sentence S_{example} is presented in Fig. 7. The lemmas were replaced with the equivalent *Synset* nodes from plWordNet after word sense disambiguation process. This process was done using our word sense disambiguation system named WoSeDon, which is based on the idea of the Personalized PageRank algorithm using an explicit knowledge base as a sense inventory (in this case plWordNet 3.1 was used).

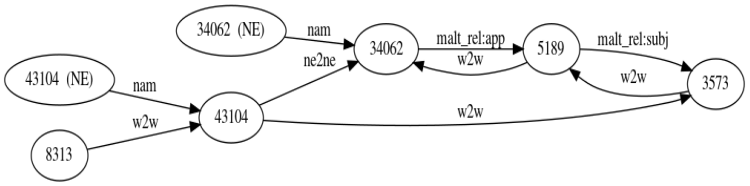

Fig. 7. Graph built for sentence S_{example} with *Synset* node type and full set of edges types

The results of the evaluation, reported in Piasecki, Kędzia and Orlińska [26], show that our system is able to reach about 55-60% of overall disambiguation precision on a mixed dataset with both monosemous and polysemous words. For a dataset with polysemous words the precision is only 38-40%, but in this case the errors seem to be deterministic, which is a positive aspect when we consider our graph similarity measures. For this particular task the performance on Wikinews documents was not evaluated. Disambiguation was performed with WoSeDon using Personalized PageRank algorithm in W2W variant (separate disambiguation for every single word in the context). Disambiguation context was limited to 3 sentences: one sentence with disambiguated word, one sentence before and one sentence after. plWordNet is a complex and rich sense inventory expressing very good coverage of Polish words with more than 284,000 lexical units (senses). In rare cases, when WoSeDon is not able to determine a sense for a word in text, this word is not represented in the graph. This type of system behaviour is particularly true only for certain, specific grammatical classes.

Third step – merging with external graph-knowledges

In **the Third Step** the constructed graphs are merged with subgraphs (one or several) extracted from an External Knowledge Graph (h EKG). Our idea is to add to the graphs built from sentences, more semantic information, extracted from EKGs. Let G be a graph with the node type t built for a sentence S during the **Second Step**, $G = (V_t E \in \{w2w, h2h, ne2ne, malt, def, srole\}$.

EKG_{plwn} is a graph built from plWordNet, where the nodes in EKG_{plwn} are the synsets from plWordNet, the edges in EKG_{plwn} are the relations from plWordNet. $EKG_{S(plwn)}$ is a subgraph of EKG_{plwn}.

EKG_{sumo} is the graph built from SUMO Ontology, where nodes represent concepts from SUMO. The edges in EKG_{sumo} correspond to SUMO relations, and $EKG_{S(sumo)}$ is a subgraph of EKG_{sumo}.

A subgraph of EKG is extracted from the source in the following way:

1. For each word w in sentence S we identify the corresponding node n_{EKG} in EKG and build a set PN_{EKG} of possible nodes.

2. For each pair of nodes ($n_{EKG,i}$, $n_{EKG,j}$) belonging to PN_{EKG} we find the shortest path sp_i from $n_{EKG,i}$ to $n_{EKG,j}$, if exists, and add sp_i to temporary graph $G_{T(S(EKG))}$. After this process $G_{S(EKG)} = G_{T(S(EKG))}$.

Using this procedure we can build three merged graphs on the basis of the graph G built in **Second Step**:

- With plWordNet, $G_{merged} = G \cup EKG_{S(plwn)}$ includes nodes of the type synset (from the **First Step**), edges built in **Second Step** and edges representing relations from the plWordNet subgraph.
- With SUMO, $G_{merged} = G \cup EKG_{S(sumo)}$ includes concept nodes from the sentence and from the subgraph of SUMO Ontology. The edges are the relations from sentence and relations from the SUMO subgraph.
- With plWordNet and SUMO, $G_{merged} = G \cup EKG_{S(plwn)} \cup EKG_{S(sumo)}$ contains full set of nodes: built in **First Step**, from plWordNet and SUMO subgraphs, i.e., edges of all types.

As a result, there are 12 possible graph types in total, i.e., four types of nodes and 3 types of merge with both EKG, namely: *Lemma lower* graph merged with $EKG_{S(sumo)}$, *Lemma PoS lower* merged with $EKG_{S(plwn)}$, *Concept* merged with $EKG_{S(sumo)}$ or *Synset* graph merged with $EKG_{S(plwn)} \cup EKG_{S(sumo)}$.

4.2.2. Similarity-based features

For each instance of relation $R_i(S_1, S_2)$, i.e., a sentence pair, from the annotated WUT CST corpus (see Section 3), 16 graphs were built for both sentences S_1 and S_2: 4 graphs with different node types in the **Second Step** and 12 graphs with combinations of every node type with both EKGs. Thus, each instance of relation R_i is assigned 16 graph-based representations of the sentence pairs $\langle S_1, S_2\rangle$: $R_i(S_1, S_2) \Rightarrow R_{ik}(G_{1k}, G_{2k})$, $k \in \{1,\ldots 16\}$. Next, we calculate eight different similarity measures between the graphs for R_i, including seven similarity measures from the literature and one proposed by us. The measures are explained further on in this section. A single instance of relation R_i from the corpus is converted into a

training vector v_i of the size 128 (16 graphs × 8 measures). As an output we received matrix $M_{N\times 128}$ where N is the number of the relations instances in corpora.

The first considered measure is the well-known Graph Edit Distance (GED) (Fernández and Valiente [7]), whose value is the minimal sum of the costs c (labelled as $\gamma(M)$) of atomic operations transforming G_1 to G_2:

$$\mathrm{GED}(G_1;G_2) = \min(\gamma(M)). \quad (1)$$

MCS (Bunke and Shearer [5]) is the ratio of the size of the Maximum Common Subgraph (MCS) of G_1 and G_2 to the size of bigger graph of (G_1 or G_2):

$$\mathrm{MCS}(G_1, G_2) = \frac{|\mathrm{mcs}(G_1, G_2)|}{\max\{|G_1|, |G_2|\}}, \quad (2)$$

Measure **WGU** (Wallis et al. [33]) depends on calculating the ratio of the size of msc G_1 and G_2 to the sum of sizes of both graphs minus msc size:

$$\mathrm{WGU}(G_1, G_2) = \frac{|\mathrm{mcs}(G_1, G_2)|}{|G_1| + |G_2| - |\mathrm{mcs}(G_1, G_2)|}. \quad (3)$$

UGU Bunke [4] is a simple measure, whose value is the difference between the sizes of G_1 and G_2 and the double size of mcs G_1 and G_2:

$$\mathrm{UGU}(G_1, G_2) = |G_1| + |G_2| - 2|\mathrm{mcs}(G_1, G_2)|. \quad (4)$$

Next measure called **MMCS** was proposed by Fernández and Valiente [7].

The MMCS value expresses the dissimilarity of graphs G_1 and G_2:

$$\mathrm{MMCS}(G_1, G_2) = |\mathrm{MCS}(G_1, G_2)| - |\mathrm{mcs}(G_1, G_2)|. \quad (5)$$

Measure **MMCSN** (Fernández and Valiente [7]) depends on calculating ratio of mcs and MCS for graphs G_1 and G_2,

$$\mathrm{MMCSN}(G_1, G_2) = \frac{|\mathrm{mcs}(G_1, G_2)|}{|\mathrm{MCS}(G_1, G_2)|}. \quad (6)$$

The last measure from literature is **Jaccard** similarity (Jaccard [9]):

$$J(G_1, G_2) = \frac{|G_1 \cap G_2|}{|G_1 \cup G_2|}. \quad (7)$$

We propose also a simple extension of the **Jaccard** measure, called **Contextual BOW**, see Equation (8). In it, the context (neighbourhood) of the node n_i from G_1 is compared with the context of node n_i in G_2. The neighbourhood of the node n in graph G is defined as input nodes $G(n)_{\mathrm{in}}$ and output nodes $G(n)_{\mathrm{out}}$:

$$N(G_1(n)) = \{G_1(n)_{\mathrm{in}} \cup G_1(n)_{\mathrm{out}}\},$$
$$N(G_2(n)) = \{G_2(n)_{\mathrm{in}} \cup G_2(n)_{\mathrm{out}}\},$$
$$S(N(G_1(n), G_2(n))) = \frac{|N(G_1(n)) \cap N(G_2(n))|}{|N(G_1(n)) \cup N(G_2(n))|},$$
$$G_{\min} = G_1 \Leftrightarrow |G_1| \le |G_2|,$$
$$G_{\min} = G_2 \Leftrightarrow |G_2| < |G_1|,$$

where $N(G_1(n))$ is the neighbourhood of node n in G_1, and $N(G_2(n))$ of the node n in G_2.

The value of CTXBowSim is calculated as:

$$\text{Sim}(G_1, G_2) = \text{CTX BowSim}(G_1, G_2) = \frac{\sum_{n \in G_{\min}}^{n} S(N(G_1(n), G_2(n)))}{|G_{\min}|}. \quad (8)$$

The similarity values are used as features during supervised learning to build a classifier. By changing the way of constructing the graphs and computing their similarity we are able to tune the classification process towards different aspects of the sentences being compared. The number of features generated for classification is dependent on the number of different graphs types, that were used to compare sentences, and the number of the applied measures for calculating their similarity. Thus, it is a combination of all node representations, all EKG sources and the applied similarity measures.

5. Results and evaluation

The corpus contains 3,469 examples annotated with one of the possible CST relations. For classification we used SVM (Support Vectors Machine), Steinwart and Christmann [32] and LMT (Logistic Model Tree) Landwehr, Hall and Frank [18]. The classifiers were evaluated according to 10-fold cross-validation scheme (Kohavi [11]).

First, the baseline set of features was tested, see Section 4.1. The classifiers were tested for the recognition of all relation types. However, the training set for the classification was highly unbalanced with respect to different relations. Table 1 shows the results for SVM and LMT and the baseline feature set. Zero values occurred for very specific relations with a small number of instances, e.g., three instances of *Citation*. Moreover, baseline features express only weak discrimination power.

Table 1. Results for the classifiers trained on the baseline feature set (lexical, syntactic, semantic)

Relation	SVM			LMT		
	P	*R*	*F*	*P*	*R*	*F*
Citation	0.000	0.000	0.000	0.000	0.000	0.000
Follow-up	0.583	0.023	0.044	0.000	0.000	0.000
Overlap	0.454	0.985	0.622	0.465	0.967	0.628
Modality	0.000	0.000	0.000	0.000	0.000	0.000
Indirect Speech	0.000	0.000	0.000	0.000	0.000	0.000
Description	0.250	0.008	0.016	0.000	0.000	0.000
Equivalence	0.000	0.000	0.000	0.000	0.000	0.000
Fulfilment	0.000	0.000	0.000	0.000	0.000	0.000
Contradiction	0.000	0.000	0.000	0.000	0.000	0.000
Summary	0.000	0.000	0.000	0.000	0.000	0.000
Historical Background	0.000	0.000	0.000	0.000	0.000	0.000
Identity	0.900	0.150	0.257	0.430	0.767	0.551
Elaboration	0.000	0.000	0.000	0.000	0.000	0.000
Subsumption	0.429	0.031	0.058	0.492	0.160	0.241
Change of Perspective	0.000	0.000	0.000	0.000	0.000	0.000
Source	0.000	0.000	0.000	0.000	0.000	0.000
No Relation	0.521	0.246	0.334	0.230	0.116	0.154
Average	0.349	0.457	0.307	0.254	0.457	0.309

In a multi-class setting, the average F-score value for SVM was 0.334 and 0.309 for LMT. Many CST relations were not recognized at all. Classifiers showed poor precision and recall in the relation detection task (*No relation* result), which means they could not decide whether a pair of sentences represents a CST link or not. The performance of the relation recognition was unsatisfactory, even for the most frequent relations including *Overlap*, *Follow-up*, *Subsumption* or *Description*.

For the graph-based approach, SVM and LMT were used again. Table 2 contains summarised results of classifiers trained with graph-based features. The performance achieved using graph-based features was better than in the previous approach. An improvement could be observed for both SVM and LMT classifiers. Only for the less frequent relations the classifiers were not able to correctly recognise the type. The average F-score value was 0.442 for SVM and 0.772 for LMT. One can notice that LMT outperforms SVM in the classification on almost every class.

Table 2 shows the results achieved on a combined set of the baseline and graph-based features. A combination of these features had a positive impact on the performance of selected classifiers. The average F-score value was increased to 0.749 for SVM and 0.817 for LMT. Our method recognised even more complex relations like *Historical Background*, *Follow-up* or *Elaboration*, with good precision and slightly lower recall. Some of the relations that occur quite rarely in our dataset were also recognised, although performance for them was still low. The corpus used for evaluation has an irregular distribution of CST relations, negatively affecting the results of classification. We can notice that for less frequent relations like *Citation*, *Modality*, *Indirect Speech* or *Contradiction*, the classifiers were not able to properly recognise types of the CST links.

Table 2. The results for a graph-based approach

Relation	SVM			LMT		
	P	*R*	*F*	*P*	*R*	*F*
Citation	0.000	0.000	0.000	1.000	0.333	0.500
Follow-up	0.965	0.180	0.303	0.772	0.853	0.811
Overlap	0.510	0.999	0.675	0.969	0.993	0.981
Modality	0.000	0.000	0.000	0.000	0.000	0.000
Indirect Speech	0.750	0.462	0.571	0.000	0.000	0.000
Description	0.578	0.070	0.125	0.556	0.739	0.634
Equivalence	0.667	0.083	0.148	0.286	0.167	0.211
Fulfilment	0.667	0.063	0.114	0.531	0.269	0.357
Contradiction	0.000	0.000	0.000	0.000	0.000	0.000
Summary	0.174	0.073	0.103	0.222	0.073	0.110
Historical Background	0.727	0.103	0.180	0.643	0.756	0.695
Identity	0.898	0.733	0.807	0.902	0.917	0.909
Elaboration	0.378	0.114	0.175	0.707	0.431	0.535
Subsumption	0.641	0.129	0.215	0.489	0.474	0.482
Change of Perspective	0.000	0.000	0.000	0.000	0.000	0.000
Source	1.000	0.820	0.901	0.813	0.520	0.634
No Relation	0.956	0.437	0.600	0.776	0.749	0.762
Average	0.620	0.544	0.448	0.771	0.786	0.772

Using different types of nodes may have a significant impact on recognition performance. By manipulating the graph structure we can show that for some relations it is easier to recognise them, if we use the appropriate node types to construct the graph. Table 3 compares all available node types in terms of their impact on recognition performance. For some relations like *Follow-up*, *Overlap* or *Description*, which are the most frequent relations in our corpus, *Lemma-PoS-Lower* graph performs slightly better than *Lemma-Lower* graph. Using the *Synset* nodes results in a significant performance drops for almost every relation. Only for *Identity*, *Elaboration* and *No Relation* we noted higher F-score values. The resulting performance drop seems to be an effect of the mistakes made by our WSD tool and our tagger inaccuracy. Using the graph with *Concept* nodes gives us the best performance for almost every relation type. The introduction of the concept nodes seem to result in some form of generalisation (e.g., individual meanings are grouped into concepts) which smooths some errors caused by the language tools.

Table 3. The results for a combined approach – basis features extended with graph-based features

Relation	SVM			LMT		
	P	*R*	*F*	*P*	*R*	*F*
Citation	0.000	0.000	0.000	0.000	0.000	0.000
Follow-up	0.800	0.967	0.876	0.964	0.961	0.962
Overlap	0.947	1.000	0.973	0.980	0.986	0.983
Modality	0.000	0.000	0.000	0.000	0.000	0.000
Indirect Speech	0.000	0.000	0.000	0.393	0.423	0.407
Description	0.551	0.728	0.627	0.613	0.707	0.657
Equivalence	0.333	0.042	0.074	0.295	0.271	0.283
Fulfilment	0.710	0.138	0.230	0.561	0.431	0.488
Contradiction	0.000	0.000	0.000	0.167	0.150	0.158
Summary	0.000	0.000	0.000	0.243	0.167	0.198
Historical Background	0.565	0.724	0.635	0.695	0.753	0.723
Identity	0.887	0.917	0.902	0.948	0.917	0.932
Elaboration	0.933	0.341	0.500	0.607	0.577	0.592
Subsumption	0.500	0.629	0.557	0.580	0.526	0.551
Change of Perspective	0.000	0.000	0.000	0.000	0.000	0.000
Source	0.800	0.160	0.267	0.818	0.720	0.766
No Relation	0.777	0.723	0.749	0.873	0.868	0.871
Avgerage	0.769	0.786	0.755	0.816	0.820	0.817

In Table 4 we also compare the performance of features constructed only for the single node type with the features combining all available node types. The acquired results confirmed our expectations and showed that features combining all types of nodes have the best discrimination ability.

The results presented for this task may suggest that we have found a simple yet effective solution for recognizing CST relations. However, acquired performance in this case may be overstated due to some specificity of our WUT CST corpus, in which instances of CST relations were pre-selected taking into account the decision agreement of our annotators, and the remaining part with uncertain instances was rejected. As it was noted earlier, a similar distribution of the relations can be observed in the CSTNews corpus (C a r d o s o et al. [6]). The authors of CSTNews

built it from news documents, i.e., the sources were very similar to those utilised in the corpus applied in this work. In M a z i e r o et al. [22] CSTNews was used to evaluate recognition methods for the refined CST model. The authors stated that their classifier outperforms other CST parsers, but a direct comparison would be inappropriate due to the differences between languages and data set structures.

Table 4. Node type impact on recognition performance (F-score) of LMT, basis features extended with graph-based features

Relation	Node type				
	Lemma	*Lemma-PoS*	*Synset*	*Concept*	*All Types*
Citation	0.000	0.000	0.000	0.000	0.000
Follow-up	0.985	**0.987**	0.908	0.925	0.962
Overlap	0.965	0.967	0.948	0.959	**0.983**
Modality	0.000	0.000	0.000	0.000	0.000
Indirect Speech	**0.478**	0.465	0.448	0.353	0.407
Description	0.618	0.636	0.595	**0.673**	0.657
Equivalence	0.228	0.202	0.227	**0.488**	0.283
Fulfilment	0.393	0.375	0.292	0.348	**0.488**
Contradiction	0.171	0.121	0.100	**0.176**	0.158
Summary	0.179	0.104	0.087	0.024	**0.198**
Historical Background	0.638	0.626	0.536	**0.745**	0.723
Identity	0.703	0.698	0.790	0.908	**0.932**
Elaboration	0.235	0.202	0.506	0.634	0.592
Subsumption	0.528	0.512	0.487	**0.580**	0.551
Change of Perspective	0.000	0.000	0.222	0.000	0.000
Source	0.723	0.622	0.514	0.000	**0.766**
No Relation	0.606	0.630	0.489	**0.897**	0.871
Average	0.762	0.758	0.749	0.794	**0.817**

6. Conslusions

In our approach a sentence *S* is represented by different graphs referring to many types of the word-level representations. It is possible to express the same sentence *S* on the morphological level (*Lemma PoS Node type*) and/or semantic level (*Synset Node type*). By merging the graphs built from *S* with some external knowledge graph, we can expand the information stored in the graph of *S* and calculate similarity between graphs more accurately. The proposed approach to build graphs is language independent and is not depended on the existence of deeper parsers.

Relations extracted from the sentence structures, i.e., *semantic roles* or *syntactic dependencies*, and lexical semantic representation assigned to words, i.e., *disambiguated senses* and *SUMO concepts*, were helpful in discriminating CST relation types. In our work we propose a method for the recognition of the full set of 17 CST relations, in contrast to the limited of subsets used in literature, e.g., in K u m a r et al. [12].

References

1. Aleixo, P., T. A. S. Pardo. Finding Related Sentences in Multiple Documents for Multidocument Discourse Parsing of Brazilian Portuguese Texts. – In: Companion Proc. of XIV Brazilian Symposium on Multimedia and the Web, WebMedia'08, New York, USA, 2008, ACM, pp. 298-303.
2. Broda, B., M. Marcińczuk, M. Maziarz, A. Radziszewski, A. Wardyński. KPWr: Towards a Free Corpus of Polish. – In: Proc. of 8th International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey, May 2012, European Language Resources Association (ELRA), Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, Stelios Piperidis, Eds.
3. Broda, B., P. Kędzia, M. Marcińczuk, A. Radziszewski, R. Ramocki, A. Wardyński. Fextor: A Feature Extraction Framework for Natural Language Processing: A Case Study in Word Sense Disambiguation, Relation Recognition and Anaphora Resolution. – In: Computational Linguistics: Applications. Adam Przepiórkowski, Maciej Piasecki, Krzysztof Jassem, Piotr Fuglewicz, Eds. Berlin, Heidelberg, Springer, 2013, pp. 41-62.
4. Bunke, H. On a Relation between Graph Edit Distance and Maximum Common Subgraph. – Pattern Recogn. Lett., Vol. **18**, August 1997, No 9, pp. 689-694.
5. Bunke, H., K. Shearer. A Graph Distance Metric Based on the Maximal Common Subgraph. – Pattern Recogn. Lett., Vol. **19**, March 1998, No 3-4, pp. 255-259.
6. Cardoso, P. C. F., E. G. Maziero, M. L. C. Jorge, E. R. M. Seno, A. Di Felippo, L. H. M. Rino, M. das G. V. Nunes, T. A. S. Pardo. CSTNews – A Discourseannotated Corpus for Single and Multi-Document Summarization of News Texts in Brazilian Portuguese. – In: Proc. of 3rd RST Brazilian Meeting, Cuiabá, Brazil, 2011, pp. 88-105.
7. Fernández, M.-L., G. Valiente. A Graph Distance Metric Combining Maximum Common Subgraph and Minimum Common Supergraph. – Pattern Recogn. Lett., Vol. **22**, May 2001, No 6-7, pp. 753-758.
8. Harary, F., R. C. Read. Is the Null-Graph a Pointless Concept? – In: Lecture Notes in Mathematics. Vol. **406**. 1974, pp. 37-44.
9. Jaccard, P. The Distribution of the Flora in the Alpine Zone. – New Phytologist, Vol. **11**, February 1912, No 2, pp. 37-50.
10. Kędzia, P., M. Maziarz. Recognizing Semantic Relations within Polish Noun Phrase: A Rule-Based Approach. – In: RANLP, 2013.
11. Kohavi, R. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. – In: Proc. of 14th International Joint Conference on Artificial Intelligence IJCAI'95, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers, Inc., Vol. **2**, pp. 1137-1143.
12. Kumar, Y. J., N. Salim, A. Hamza, A. Abuobieda. Automatic Identification of Cross-Document. – Structural Relationships, Vol. **7**, 2012, pp. 26-29.
13. Kumar, Y. J., N. Salim, B. Raza. Cross-Document Structural Relationship Identification Using Supervised Machine Learning. – Appl. Soft Comput., Vol. **12**, October 2012, No 10, pp. 3124-3131.
14. Kumar, Y. J., N. Salim, A. Abuobieda, A. T. Albaham. Multi Document Summarization Based on News Components Using Fuzzy Cross-Document Relations. – Applied Soft Computing, Vol. **21**, 2014, pp. 265-279.
15. Kędzia, P., M. Piasecki. Ruled-Based, Interlingual Motivated Mapping of plWordNet onto SUMO Ontology. – In: Proc. of 9th International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, 26-31 May 2014, Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asunción Moreno, Jan Odijk, Stelios Piperidis, Eds., pp. 4351-4358.
16. Kędzia, P., M. Piasecki, J. Kocoń, A. Indyka-Piasecka. Distributionally Extended Network-Based Word Sense Disambiguation in Semantic Clustering of Polish Texts. – IERI Procedia, Vol. **10** (Complete), 2014, pp. 38-44.

17. Kędzia, P., M. Piasecki, M. Orlińska. Word Sense Disambiguation Based on Large Scale Polish CLARIN Heterogeneous Lexical Resources. – Cognitive Studies/Études Cognitives, Vol. **15**, 2015, pp. 269-292.
URL: https://ispan.waw.pl/journals/index.php/cs-ec/article/download/cs.2015.019/1765
18. Landwehr, N., M. Hall, E. Frank. Logistic Model Trees. – Machine Learning, Vol. **59**, 2005, No 1, pp. 161-205. ISSN 1573-0565.
19. Marcińczuk, M. Automatic Construction of Complex Features in Conditional Random Fields for Named Entities Recognition. – In: RANLP, 2015.
20. Marcińczuk, M., J. Kocoń, M. Janicki. Liner2 – A Customizable Framework for Proper Names Recognition for Polish. – In: Intelligent Tools for Building a Scientific Information Platform, Robert Bembenik, Łukasz Skonieczny, Henryk Rybiński, Marzena Kryszkiewicz, Marek Niezgódka, Eds., 2013, pp. 231-253.
21. Maziarz, M., M. Piasecki, E. Rudnicka, S. Szpakowicz, P. Kędzia. plWordNet 3.0 – A Comprehensive Lexical-Semantic Resource. – In: Proc. of 26th International Conference on Computational Linguistics, COLING 2016, Technical Papers, 11-16 December 2016, Osaka, Japan, pp. 2259-2268.
22. Maziero, E. G., M. L. D.-R. C. Jorge, T. A. S. Pardo. Revisiting Cross-Document Structure Theory for Multidocument Discourse Parsing. – Inf. Process. Manage, Vol. **50**, March 2014, No 2, pp. 297-314.
23. Nivre, J., J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov, E. Marsi. MaltParser: A Language-Independent System for Data-Driven Dependency Parsing. – Natural Language Engineering, Vol. **13**, 2007, No 2, pp. 95-135.
24. Pease, A. Ontology: A Practical Guide. Angwin, CA, Articulate Software Press, 2011.
25. Piasecki, M., S. Szpakowicz, B. Broda. A Wordnet from the Ground Up. – Oficyna Wydawnicza Politechniki Wroclawskiej, Wrocław, 2009.
26. Piasecki, M., P. Kędzia, M. Orlińska. plWordNet in Word Sense Disambiguation Task. – In: Proc. of 8th Global Wordnet Conference (GWC'16), Bucharest, 27-30 January 2016, Osaka, Japan, pp. 280-290.
27. Radev, D. R. A Common Theory of Information Fusion from Multiple Text Sources Step One: Cross-Document Structure. – In: Proc. of 1st SIGdial Workshop on Discourse and Dialogue, SIGDIAL'00, Association for Computational Linguistics, Stroudsburg, PA, USA, Vol. **10**, 2000, pp. 74-83.
28. Radev, D. R., J. Otterbacher, Z. Zhang. Cst Bank: A Corpus for the Study of Cross-Document Structural Relationships. – In: European Language Resources Association, LREC, 2004.
29. Radziszewski, A. A Tiered CRF Tagger for Polish. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 215-230.
https://doi.org/10.1007/978-3-642-35647-6_16
30. Radziszewski, A., A. Pawlaczek. Language Processing and Intelligent Information Systems. – In: Proc. of 20th International Conference, IIS 2013, Warsaw, Poland, 17-18 June 2013. Chapter Incorporating Head Recognition into a CRF Chunker, Berlin, Heidelberg, Springer, 2013, pp. 22-27.
31. Radziszewski, A., A. Wardyński, T. Śniatowski. WCCL: A Morpho-Syntactic Feature Toolkit. – In: Proc. of Balto-Slavonic Natural Language Processing Workshop (BSNLP'11), Springer, 2011.
32. Steinwart, I., A. Christmann. Support Vector Machines. First Edition. Springer Publishing Company, Inc., 2008.
33. Wallis, W. D., P. Shoubridge, M. Kraetz, D. Ray. Graph Distances Using Graph Union. – Pattern Recogn. Lett., Vol. **22**, May 2001, No 6-7, pp. 701-704.
34. Woliński, M. Morfeusz – A Practical Tool for the Morphological Analysis of Polish. – In: Mieczysław A. Kłopotek, Sławomir T. Wierzchoń, Krzysztof Trojanowski, Eds. Intelligent Information Processing and Web Mining, Advances in Soft Computing, Berlin, Springer, 2006, pp. 503-512.
35. Wróblewska, A., M. Woliński. Preliminary Experiments in Polish Dependency Parsing. Berlin, Heidelberg, Springer, 2012, pp. 279-292.

36. Wróblewska, A. Polish Dependency Parser Trained on an Automatically Induced Dependency Bank. Ph.D. Dissertation, Institute of Computer Science, Polish Academy of Sciences, Warsaw, 2014.
37. Zahri, N. A. H. B., F. Fukumoto. Multi-Document Summarization Using Link Analysis Based on Rhetorical Relations between Sentences, Berlin, Heidelberg, Springer, 2011, pp. 328-338.
38. Zhang, Z., D. Radev. Combining Labeled and Unlabeled Data for Learning Cross-Document Structural Relationships. – In: Proc. of 1st International Joint Conference on Natural Language Processing, Berlin, Heidelberg, Springer, 2005, pp. 32-41.
39. Zhang, Z., J. Otterbacher, D. Radev. Learning Cross-Document Structural Relationships Using Boosting. – In: Proc. of 12th International Conference on Information and Knowledge Management (CIKM'03), ACM, New York, USA, 2003, pp. 124-130.

Received 20.10.2017; *Second Version 6.12.2017*; *accepted 31.01.2018*