IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 2, 2020, accepted July 23, 2020, date of current version August 27, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3014862

High-Precision Biomedical Relation Extraction for Reducing Human Curation Efforts in Industrial Applications

ALAN RAMPONI[1,2], STEFANO GIAMPICCOLO[1], DANILO TOMASONI[1], CORRADO PRIAMI[1,3], AND ROSARIO LOMBARDO[1]

[1]Fondazione the Microsoft Research, University of Trento Centre for Computational and Systems Biology (COSBI), 38068 Rovereto, Italy
[2]Department of Information Engineering and Computer Science, University of Trento, 38123 Povo, Italy
[3]Department of Computer Science, University of Pisa, 56127 Pisa, Italy

Corresponding author: Rosario Lombardo (e-mail: lombardo@cosbi.eu).

ABSTRACT The body of biomedical literature is growing at an unprecedented rate, exceeding the ability of researchers to make effective use of this knowledge-rich amount of information. This growth has created interest in biomedical relation extraction approaches to extract domain-specific knowledge for diverse applications. Despite the great progress in the techniques, the retrieved evidence still needs to undergo a time-consuming manual curation process to be truly useful. Most relation extraction systems have been conceived in the context of Shared Tasks, with the goal of maximizing the F1 score on restricted, domain-specific test sets. However, in industrial applications relations typically serve as input to a pipeline of biologically driven analyses; as a result, highly precise extractions are central for cutting down the manual curation effort, thus to translate the research evidence into practice smoothly and reliably. In this paper, we present a highly precise relation extraction system designed to reduce human curation efforts. The engine is made up of sophisticated rules that leverage linguistic aspects of the texts rather than sticking on application-specific training data. As a result, the system could be applied to diverse needs. Experiments on gold-standard corpora show that the system achieves the highest precision compared with previous rule-based, kernel-based, and neural approaches, while maintaining a F1 score comparable or superior to other methods. To show the usefulness of our approach in industrial scenarios, we finally present a case study on the mTOR pathway, showing how it could be applied on a large-scale.

INDEX TERMS Biomedical text mining, information extraction, natural language processing, relation extraction, syntactic dependencies.

I. INTRODUCTION

In the last 30 years we have positively observed a rapidly growing body of biomedical literature. As a consequence, it is more and more difficult for researchers to keep pace with the advances in their fields. Indeed, it has been recently shown that one would have to examine 27 papers per day from 130 previously scanned journals to stay up to date with the literature about a single, specific disease [1]. Such a large volume of written biomedical knowledge is becoming increasingly available in the form of electronic data resources such as digital libraries and biomedical databases. The largest bibliographic archives such as PubMed [2] and PubMed Central (PMC) [3] give access to a total of over 31 million abstracts and 6.3 million full text documents that are currently growing at a double-exponential rate. Since researchers struggle to cope with this amount of data, the development of effective biomedical text mining systems has become increasingly important to allow them to dig through undiscovered knowledge.

A variety of text mining tools have been developed over the last two decades. Efforts by the US National Library of Medicine have led to the well-known PubMed search service which allows users to browse research publications filtered according to user queries including concepts specified with manually curated MeSH terms [4]. Systems such as FACTA [5] and Polysearch2 [6], [7] have also been conceived to retrieve relevant information exploiting the co-occurrence of

The associate editor coordinating the review of this manuscript and approving it for publication was Arianna Dulizia.

the concepts of interest. However, assuming concepts mentioned together to be related typically leads to a lot of irrelevant results. Natural Language Processing (NLP) techniques have begun to be explored in the last two decades in order to effectively derive meaning from human language in a deeper useful way. Particularly, Relation Extraction (RE) has attracted a lot of interest as a valuable tool ranging from the population of knowledge bases to the construction of biochemical pathways [8]. To encourage the development of highly performing relation extraction systems, several community challenges (i.e., Shared Tasks) [9] have been designed. For instance, biomedical relation extraction has been employed to answer research questions ranging from the identification of protein-protein interactions [10], [11], gene-disease associations [12]–[15], adverse drug events [16], [17], and protein subcellular localization [18]. Many systems have been designed to deal with the peculiarities of the specific application domain because of the great complexity and diversity of topics included in the biomedical literature, falling short when dealing with different research questions. As a matter of fact, most systems have been proposed in the context of Shared Tasks, in which the focus is on improving the harmonic mean of precision and recall (i.e., the F1 score) on specific test data, rather than providing a highly precise extraction of information that cuts down the need of a human manual curation. Indeed, the results of relation extraction systems still need to undergo a manual scrutiny by field experts in order to make the information ready to be exploited. This resource-demanding manual scrutiny should ideally be avoided in real-world contexts, where biomedical relation extraction is the first step of a complex pipeline of biologically driven analyses which requires highly precise relations in order to produce reliable insights. This is even more important because of the rapidly growing body of biomedical literature, which calls for frequent updates of the extracted evidence during a project life cycle. Highly precise relation extraction results, with a satisfactory recall, are thus crucial in real-world scenarios to smoothly translate the extracted information into actionable knowledge.

In this paper, we present a relation extraction system designed to extract highly precise semantic relationships from biomedical texts without the need of training data. Our approach is based on a sequence of NLP syntactic modules, and a novel dependency tree based relation engine which captures relations by means of syntactic rules based on common linguistic patterns. As a result, our system could be applied to different corpora without relying on application specific biomedical relation instances. The highly precise results largely limit the need for human manual curation, allowing scientists to quickly keep abreast of novel discoveries and thus to drive an effective research.[1]

[1]A docker container is available at: https://www.cosbi.eu/research/prototypes/biomedical_knowledge_extraction.

The paper is organized as follows. Section II lists related work in the field. Section III describes the methods of our system, going through the natural language processing analysis and the relation extraction engine. Section IV presents the results of our system showing the quality of the method with respect to well-established gold-standard corpora and recent approaches. Also, a detailed error analysis, current limitations, and room for improvements are discussed. Section V outlines a case study to show how our system could be effectively applied on large-scale industrial scenarios, whereas conclusions are in Section VI.

II. RELATED WORK

A variety of methods has been adopted for biomedical relation extraction. These approaches can be mainly divided into three categories: rule-based methods, feature- and kernel-based methods, and neural methods. Rule-based approaches typically make use of linguistically-motivated patterns on dependency parse trees or surface words in order to capture semantic relationships. Fundel *et al.* [19] showed how a small number of carefully designed rules based on the shortest dependency path (SDP) between two examined entities produces fairly good results. Yu *et al.* [20] exploited dependency parse trees and a flexible pattern matching scheme, enriching the system with a decision tree classifier. Diverse syntactic and orthography features have been extensively used in feature- and kernel-based methods. Phan *et al.* [21] proposed an automatic feature selection method based on the contribution levels of different feature groups, followed by a k-nearest neighbor (k-NN) classifier. A variety of kernel-based methods have been proposed too, ranging from the walk-weighted subsequence kernel [22] to a combination of kernels based on different parsers [23]. Other kernel-based approaches for biomedical relation extraction include a linguistic pattern-aware dependency tree kernel combined with a tree kernel [24], a convolution tree kernel [25], and a distributed smoothed tree kernel combined with a feature kernel [26]. In the rising wave of deep learning, Zhao *et al.* [27] proposed a deep multi-layer neural network for the task. More recent neural methods use Recurrent Neural Networks (RNNs), including Bidirectional Long Short-Term Memory (LSTM) and tree LSTM networks, and Convolutional Neural Networks (CNNs). Zhang *et al.* [28] showed how leveraging the complementary advantages of RNNs and CNNs in a combined hybrid model improves biomedical relation extraction. Yadav *et al.* [29] experimented with a bidirectional LSTM network with an attention mechanism, exploiting word sequences and the shortest dependency path between the entities, whereas Zhang *et al.* [30] introduced a residual CNN to tackle the task. Ahmed *et al.* [31] exploited a tree LSTM network using a structured attention architecture, showing how the attention mechanism improves the performance in relation extraction. A recent research line in NLP include the Transformer, an encoder-decoder architecture which dispenses with convolutions and recurrence, being based solely on an attention mechanism [32].

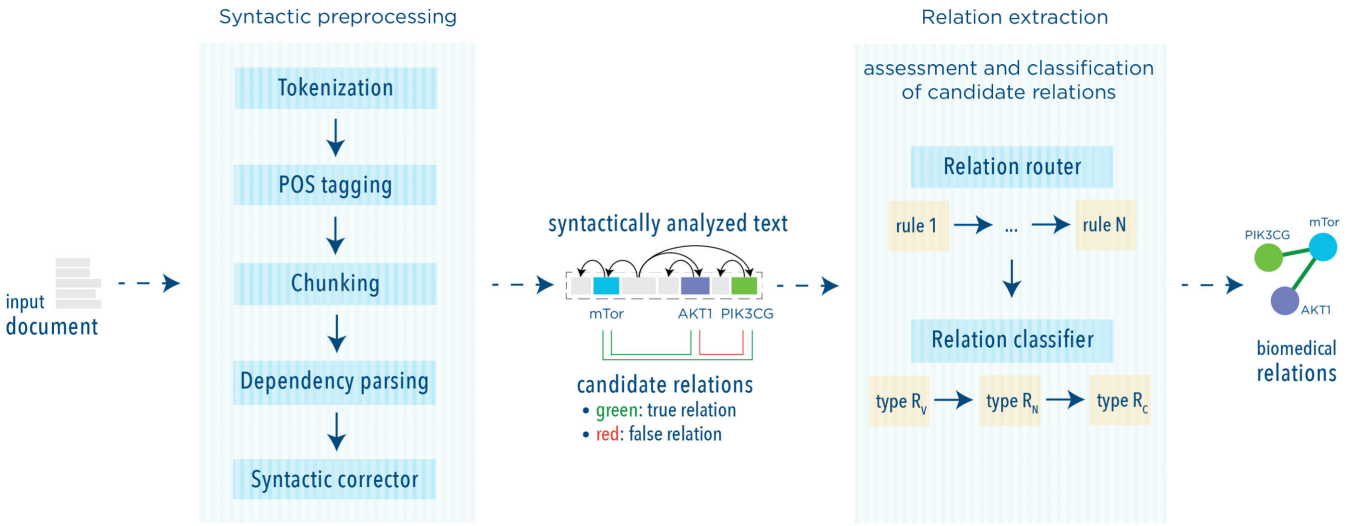


FIGURE 1. In our biomedical relation extraction approach each input document is firstly analyzed by syntactic preprocessing modules (i.e., tokenizer*, POS tagger, chunker*, dependency parser, and syntactic corrector*). The resulting syntactic dependency parse tree and token annotations, along with candidate entity pairs, are analyzed by a relation router to detect candidate relations. Actual relations are finally identified by a relation classifier, powered with pattern matching rules on the dependency tree. *Custom implementation of preprocessing components.

This architecture is the core of pre-trained language models such as BERT (Bidirectional Encoder Representations from Transformers) [33], and its adaptively pre-trained variants for biomedical texts, namely BioBERT [34] and SciBERT [35]. Despite the recent advances in deep learning based techniques, we rely on carefully designed syntactic rules on dependency parse trees in order to avoid being dependent on labeled data, and to be able to reuse our system in diverse industrial scenarios. The most similar approach to our work is thus represented by the work by Fundel *et al.* [19].

III. METHODS

The system is designed to extract highly precise relational information from input texts. In Fig. 1 we schematically show our approach to relation extraction, that includes two stages: (i) text preprocessing, in which a sequence of natural language processing modules are applied to texts (Section III-A), and (ii) relation extraction, in which relationships between entities are identified and classified (Section III-B).

A. SYNTACTIC PREPROCESSING

A pipeline of natural language processing modules is needed in order to provide the relation extraction engine the information needed to extract highly precise semantic associations. We present them in the following.

1) TOKENIZATION

The raw text is separated into tokens using the spaCy[2] tokenizer. We customize it to segment text units also on punctuation (e.g., hyphens, slashes, etc.) by means of regular expressions. This fine-grained approach to tokenization originates from the observation that not all the symbol-separated tokens are the smallest units of information to work with. For instance, ''IL6-induced atrophy'' is typically divided in two tokens (''IL6-induced'' and ''atrophy''). However, ''IL6-induced'' implicitly encodes relational information that is eventually desirable to analyze.

2) PART-OF-SPEECH TAGGING

Each token is assigned a label describing its part-of-speech (POS) at two different granularities: a coarse-grained one (from Universal POS tags[3]) and a fine-grained one (from Penn Treebank tags[4]). We use the spaCy *en_core_web_lg* neural model for POS tagging. These labels serve to both the chunking and the syntactic dependency parsing steps.

3) CHUNKING

Our fine-grained approach to tokenization allows the system to ultimately merge only the tokens that together form a self-contained chunk of information. For instance, given the set of tokens $T = \{(, AKT,), -, 1\}$, they are part of the same concept ''(AKT)-1'', hence it is desirable to merge them into a single token. To the goal, we designed sequence patterns based on the orthography, and on the fine- and coarse-grained POS tags of the tokens. The resulting token stores all attributes of its original constituents (POS tags, lemmas, surface text, etc.). The list of patterns is in Table 1, together with common examples. We designed this module to:

1) reduce potential errors in syntactic parsing in case of long and articulated texts;
2) process easily multi-token words (e.g., ''Interleukin 6''), frequent in biomedical texts.

[2]www.spacy.io

[3]http://universaldependencies.org/u/pos/

[4]https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

TABLE 1. The chunking patterns used by the system. Each token T_i in a candidate sequence $T_1 \dots T_n$ must satisfy some pattern restrictions on the orthography and part-of-speech levels in order to be merged. Underlined tokens are the triggers of a pattern, which have to meet their restrictions in order to proceed. The rest of the sequence tokens are thus subsequently checked for restrictions. If all the pattern restrictions are satisfied, the merging rule is applied.

Pattern name	Token sequence	Pattern restrictions	Examples
PLUS_MATCHER$_1$	$\underline{T_1}T_2$	$T_1(tag) \in \{NN, NNS, NNP, NNPS\}$ $T_2(orth) \in \{+\}$	CD19+ CD20 +
PLUS_MATCHER$_2$	$\underline{T_1}T_2T_3T_4$	$T_1(tag) \in \{NN, NNS, NNP, NNPS\}$ $T_2(tag) \in \{-LRB-\}$ $T_3(orth) \in \{+\}$ $T_4(tag) \in \{-RRB-\}$	CD19(+) CD20 (+)
HYPHEN_MATCHER$_1$	$\underline{T_1}T_2T_3$	$T_1(tag) \in \{NN, NNS, NNP, NNPS\}$ $T_2(tag) \in \{HYPH\}$	IL-6 AKT- 1
HYPHEN_MATCHER$_2$	$T_1\underline{T_2}T_3T_4T_5$	$T_1(tag) \in \{-LRB-\}$ $T_2(pos) \in \{PROPN\}$ $T_3(tag) \in \{-RRB-\}$ $T_4(orth) \in \{-\}$ $T_5(tag) \in \{CD\}$	(IL)-6 (AKT) - 1
ADJ_NN_MATCHER	$\underline{[T_1, ..., T_k]}[T_{(k+1)}, ..., T_n]$	$T_A(tag) \in \{JJ, JJR, JJS\}$ $T_B(pos) \in \{NOUN, PROPN\}$ where $T_A \in \{T_1, ..., T_k\}, T_B \in \{T_{(k+1)}, ..., T_n\}$	Primary cell Tumor suppressor protein
ADJ_COMPOUND_MATCHER	$\underline{[T_1, ..., T_k]}[T_{(k+1)}, ..., T_n]$	$T(pos) \in \{ADJ\}$ where $T \in \{T_1, ..., T_n\}$	Young adult
NN_COMPOUND_MATCHER	$\underline{[T_1, ..., T_k]}[T_{(k+1)}, ..., T_n]$	$T(pos) \in \{NOUN, PROPN\}$ where $T \in \{T_1, ..., T_n\}$	Cell antibody

For instance, the units the chunker produces for the following sentence are presented inside brackets:

*[YtxH] [and] [YvyD] [are] [induced] [after] **[phosphate starvation]** [in] [the] **[wild type]** [in] [a] **[sigma(B)]**[-][dependent] [manner]*

where in bold are the merged tokens. The number of chunks decreases from 21 to 16, allowing an easier parsing process, and multi-token words are produced. For simplicity, we hereafter refer to tokens and chunks indistinctly.

4) SYNTACTIC DEPENDENCY PARSING

A syntactic dependency parse tree of the text is built using the spaCy non-monotonic transition-based parser. We chose to rely on the spaCy parser since it has been benchmarked to be the fastest to date,[5] and thus it fully meets industrial requirements. The grammatical dependencies (hereafter, *edges*) of the tokens or chunks (hereafter, *nodes*) are drawn from the CLEAR tag set for dependency parsing.[6]

5) SYNTACTIC CORRECTOR

The predicted POS tags and grammatical dependencies that are assigned to tokens are not always correct. Correcting POS tags or parse trees as a whole is an hard problem; however, some wrong labels can be easily detected. As a consequence, for the most trivial errors we automatically correct the labels, whereas in more complex cases we label the sentence as unreliable to avoid false positives in the relation extraction phase. The following corrections are applied:

- tokens that are heads of a direct object (*dobj*) or a nominal subject (*nsubj*) having a coarse-grained POS tag different from *verb* are assigned *verb* as a POS tag;
- tokens that are heads of an adjectival modifier (*amod*) having *verb* as a coarse-grained POS tag are assigned *adj* as a POS tag, since they are in most cases past participles used as adjectives.

B. RELATION EXTRACTION

The syntactic preprocessing provides the information needed to extract biomedical relationships between entities from text. Following previous work in biomedical relation extraction, we assume entities are given. We rely on syntactic rules, in order to have a single system that can be applied to diverse corpora, completely removing the need of training data. Thus we fully exploit the dependency parse tree and the syntactic information encoded to each token. Our strategy involves a routing phase to detect candidate relations (Section III-B.1), and a classification phase to assess the relations, assigning them an *effector* and an *effectee* roles (Section III-B.2).

1) RELATION ROUTER

We analyze the minimum path of the dependency parse tree between entities to assess if the path is eligible for representing a candidate relation pair. We devise several rules to the goal, based on common linguistic constructs. The process of routing a syntactic path involves both the analysis of crossed edges (i.e., dependency relations) and node attributes (e.g., lemma, POS tags, etc.). Fig. 2 summarizes the workflow. The router stops immediately labeling the candidate relation pair as negative if one of the following conditions is met:

[5]https://spacy.io/usage/facts-figures#benchmarks
[6]https://github.com/clir/clearnlp-guidelines/

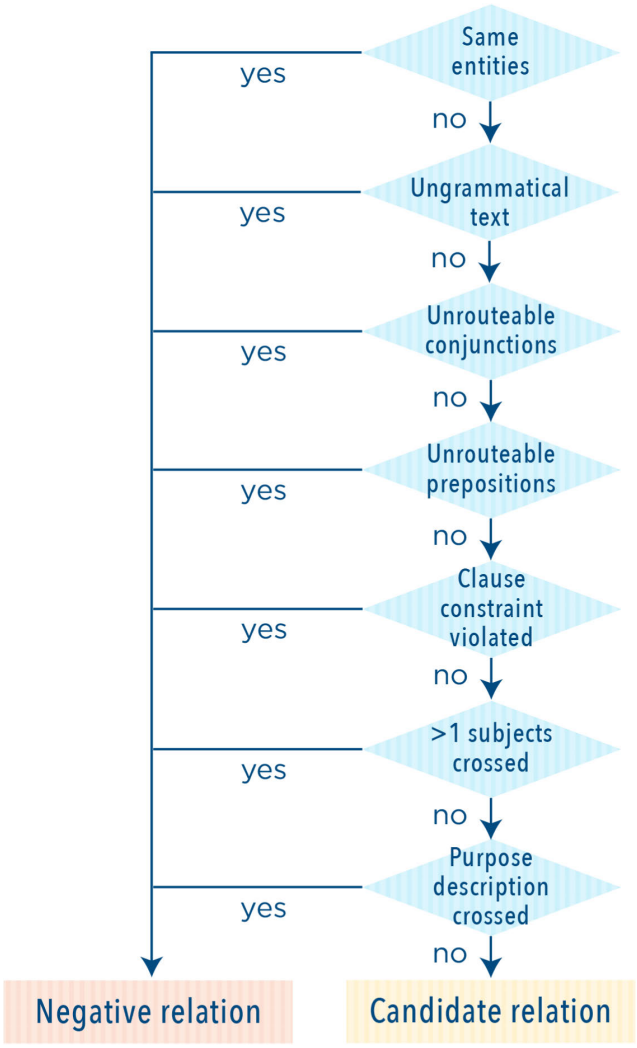


FIGURE 2. The logic of the relation router. Rhombus shapes indicate tested conditions, while arrows indicates the router flow. If all the conditions are negative, the entity pair is considered a relation candidate. Otherwise, the entity pair is labeled as a negative relation instance.

1) **Same entities.** If the lemmas of entities are the same, the candidate pair is labeled as negative;
2) **Ungrammatical text.** In the case the input text has no verb if not in subordinate clauses, the pair is considered unreliable and thus labeled as a negative instance;
3) **Unrouteable conjunctions.** If conjunctions introducing subordinate or coordinate prepositions are met (i.e., *but*, *whereas*, *if*, *therefore*, and *while*), the entities are unlikely to be related, thus the pair is negative;
4) **Unrouteable prepositions.** The prepositions *if*, *therefore*, *during*, *despite*, *from*, and *at* typically introduce phrases that specify where – or when – a specific event occurs – or has occurred. If one of these prepositions is found in the path, the candidate pair is unlikely to be a relation and thus discarded (i.e., labeled as negative);
5) **Clause routing constraint.** Sentences in the biomedical literature are complex and articulated, with one or more coordinate and subordinate clauses. Entities in different clauses could be in a relation, but only under some conditions. We allow the router to cross a clause only if the target clause has no explicit subject dependency, and if the final path has exactly one subject. Otherwise, we consider the pair a negative instance;
6) **More than one subject crossed.** If more than a subject dependency relation is crossed we label the relation pair as negative, because the minimum path is typically crossing semantically independent phrases or clauses. For instance, in the sentence "A causes B and C triggers a D-reaction", the entity A is not related to the entity D;
7) **Purpose-description statements.** Some sentences express a broad research purpose (e.g., "In this paper we aim to demonstrate that tuberculosis could be prevented by vaccines."), instead of actual relations. When crossing the path between entities, the lemmas of the tokens is thus compared to a list of purpose-related words (Supplementary File 1, "`purpose_words`"). If a match is found, the pair is labeled as negative.

While crossing the path, the relation router also checks if the relation is affirmed or negated. This is particularly useful to detect actual associations for real-world use. Negations are detected using the following rules:

1) **Negative auxiliary.** A crossed token node is incident to an edge having a negation modifier dependency tag (*neg*), or is adjacent to a token node with *no* lemma;
2) **Negative verb.** One of the crossed verbs belongs to a negative meaning verb list (Supplementary File 1, Section "`negation_verbs`");
3) **Negative adverb.** A crossed token node is incident to an edge having a negation adverb as target (Supplementary File 1, Section "`negation_adverbs`");
4) **Negative noun.** One of the crossed nouns belongs to a negative meaning noun list (Supplementary File 1, Section "`negation_nouns`").
5) **Negative adjective.** One of the crossed adjectives belongs to a negative meaning adjective list (Supplementary File 1, Section "`negation_adjectives`").

If the relation router navigates the whole path between the two entities without any of the routing conditions is met, the pair is considered a relation candidate and is analyzed by the relation classifier (Section III-B.2).

2) RELATION CLASSIFIER

The relation classifier analyzes the relation candidates the router identified, assigning the entities the *effector* and the *effectee* roles. We identified three categories of linguistic constructs that are typically used to express semantic relations in the English language. The categories are the following:

- **Relation expressed by a verb** (R_V). A generalized version of the *effector-relation-effectee* rule proposed in [19] that we enhanced to capture constructs of the form:

  $entity_A$-*[phrase]-verb-[phrase]-*$entity_B$

where a *phrase* can appear zero, one, or multiple times. As a result, the rule matches elaborate statements with interleaved phrases such as "A plays a big role in B assimilation" or "abundance of A causes B degradation", and not only triples of the form *entity$_A$-verb-entity$_B$*.

- **Relation expressed by a nominalization or a participle** (R_N). Associations in the biomedical literature are often expressed by nominalizations or participles. We thus employ the following rules:

  (1) *nominalization-of-entity$_A$-by-entity$_B$*
  Example: "Activation of A by B"

  (2) *nominalization-between-entity$_A$-and-entity$_B$*
  Example: "Relation between A and B"

  (3) *nominalization-of-entity$_A$-on-entity$_B$*
  Example: "Effect of A on B"

  (4) *entity$_A$-participle-entity$_B$*
  Example: "A-activated B", "A-dependent B"

  While rules (1) and (2) are inspired by the *relation-of-effectee-by-effector* and *relation-between-effector-and-effectee* proposed in [19], the rule (3) widens the scope of rule (1), and rule (4) allows the system to effectively handle nominalized adjectives expressing relations.
- **Relation expressed by a conjunction** (R_C). This category is designed to capture relations of entities that act together to do something, which are typically both subjects of a statement. We use the following pattern:

  entity$_A$-conjunction-entity$_B$-verb
  Example: "A and B form a complex"

As a result, if the path between the entities contains a verb, we consider the candidate relation pair as a R_V relation. The verb found in the path is considered the verb for the relation, and if multiple verbs are found, we take the last one in the text order. To assign the roles to the entities, we look at the verb voice. If the voice is active, the entity that appears first in the sentence is labeled as the effector, while the second one is labeled as the effectee. Otherwise, the first entity is labeled as the effectee, and the second entity as the effector.

In the case no verb is found in the crossed path, but it contains (a) a past participle,[7] (b) an adjective ending in "ent" (e.g., A-dependent B), or (c) a nominalized verb, we consider the candidate pair as a R_N candidate. Additionally, we have to focus on the types of the edges crossed. During the routing we allow many edge types to be crossed, but a lot of them only exist in verb-expressed relations. Since a R_N relation represents a more compact connection between the entities, it should not contain verbs (if not the participle form), nor both subjects and objects. To model this additional restriction in terms of edge types and paths, we check whether the minimum path between the two entities only contains certain types of grammatical dependencies. Beyond links expressed by conjunctions and prepositions, only modifiers, compounds, and appositions should exist (i.e., *npadvmod*, *amod*, *compound*, *appos*, *punct*, *prep*, *pobj*, or *conj*). If condition (a) or (b) is satisfied, the effector and effectee roles are assigned according to the text order, whereas if condition (c) is met, roles are assigned by analyzing the preposition connecting the nominalization and the entities. Specifically, the effector is the entity that does not have a preposition *or*, *by* as ancestor, whereas the effectee is the entity that has a preposition amongst *on*, *of*, or *with* as ancestor.

If the crossed path only contains a conjunction, the remaining part of the text is analyzed for R_C relations. We check whether the top-level node of the path is incident to a verb node. In such case, we check if the verb lemma is *interact* or *form*, and if so, we consider the relation as a R_C type.[8] Note that in the R_C category the effector and effectee roles are not needed since both entities are interacting as both effectors.

Lastly, if all R_V, R_N, and R_C categories are not satisfied, the candidate relation pair is labeled as negative.

[7]This holds under some limitations: it should be incident to an edge with a *npadvmod* or an *amod* dependency relation.

[8]In contrast to R_V and R_N relation categories, we here look at the whole dependency tree, without restricting the focus to the minimum path.

TABLE 2. Statistics of the benchmark corpora.

	LLL	IEPA	HPRD50
Positive relations	164	335	163
Negative relations	166	482	270
Sentences	77	486	145

IV. RESULTS AND DISCUSSION

We evaluate our relation extraction method on different benchmark corpora annotated for biomedical relations: LLL [36], IEPA [37], and HPRD50 [19]. The corpora are about different topics in biomedicine, thus they represent a good evaluation benckmark for our system for diverse real-world applications. In particular, LLL is a corpus about the model bacterium *Bacillus subtilis*, focused on gene transcription and sporulation; HPRD50 is about regulatory relations, direct physical interactions and modifications on documents from the Human Protein Reference Database [38]; and IEPA is a corpus focused on interactions between a restricted set of biochemicals (e.g., insulin, oxytoxin, leptin, etc.). Relations between entities are annotated within the sentence boundaries, and entities offsets are provided with the raw texts. Given a set of entities $\{e_1, e_2, \ldots, e_n\} \in E$ belonging to an input sentence S, we generate $\binom{n}{2}$ candidate relation instances (if $n \geq 2$) for the sentence S. Following previous work, negative instances are represented by pairs that are not annotated as relations in the corpora. The statistics of the corpora are summarized in Table 2.

For the sake of comparison to previous work, we evaluate our relation extraction method using precision (1), recall (2), and F1 score (3):

$$precision = \frac{TP}{TP + FP} \tag{1}$$

TABLE 3. Performance comparison of our system with other approaches on benchmark corpora. Precision (P), Recall (R), and F1 score (F1) are shown by percentage rounded with a single decimal. Best results for each metric are highlighted in bold. GRGT = Grammatical Relationship Graph for Triplets; O2G = Optimized combination of 2 Groups with the best contribution levels; *k*-NN = *k*-Nearest Neighbor; LPTK = Linguistic Pattern-aware dependency Tree Kernel; DSTK = Distributed Smoothed Tree Kernel; RNN = Recurrent Neural Network; CNN = Convolutional Neural Network; SDP = Shortest Dependency Path; LSTM = Long Short-Term Memory; Bi-LSTM = Bidirectional Long Short-Term Memory. *Original implementation we fine-tuned on each corpus.

Method	LLL			HPRD50			IEPA		
	P	R	F1	P	R	F1	P	R	F1
Dependency tree rules [19]	79.0	85.0	82.0	79.0	78.0	78.0	-	-	-
GRGT (matching rules & decision tree) [20]	91.2	77.1	83.6	86.5	50.8	64.0	91.0	63.6	74.9
O2G (feature selection & *k*-NN) [21]	74.7	82.2	76.5	73.0	74.3	72.6	68.1	71.3	69.5
Walk-weighted subsequence kernel [22]	76.9	91.2	82.4	66.7	69.2	67.8	73.8	71.8	72.9
Multiple kernels [23]	77.6	86.0	80.1	68.5	76.1	70.9	67.5	78.6	71.7
LPTK (patterns & tree kernel) [24]	78.9	72.1	75.3	72.7	62.2	67.1	74.8	66.1	70.2
Convolution tree kernel [25]	73.2	89.6	80.6	63.8	81.2	71.5	62.5	83.3	71.4
DSTK & feature kernel [26]	87.3	91.2	89.2	76.3	84.2	80.0	75.9	85.2	80.2
Deep neural network [27]	80.7	84.4	81.0	58.7	92.4	71.3	68.7	83.5	74.2
Combined RNN and CNN [28]	76.6	96.1	85.2	75.1	76.4	75.6	73.2	84.4	78.2
SDP-based Bi-LSTM with attention [29]	84.2	83.6	83.9	79.9	77.6	78.7	77.0	75.6	76.3
Residual CNN [30]	80.5	87.2	83.2	74.9	82.8	77.7	71.6	80.6	75.5
Tree LSTM [31]	85.3	84.9	84.8	82.4	82.8	82.0	77.0	76.7	76.4
Tree LSTM with structured attention [31]	84.8	84.3	84.2	81.7	82.3	81.3	78.6	78.7	78.5
BioBERT* [34]	87.0	91.8	89.4	83.1	81.8	**82.5**	79.7	82.2	80.9
SciBERT* [35]	87.3	94.0	**90.5**	79.1	76.9	78.0	81.6	77.7	79.6
Ours	**93.2**	73.1	81.9	**90.7**	70.8	79.5	**91.7**	72.8	**81.1**

$$recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1\_score = \frac{2 * precision * recall}{precision + recall} \tag{3}$$

where *TP* is the number of true positives, *FP* is the number of false positives, and *FN* is the number of false negatives. However, we are mainly interested in the precision metric, motivated by a real-world application of the system.

We compared our method to existing methods in literature, including rule-based approaches [19], [20], feature- and kernel-based approaches [21]–[26], and neural network approaches [27]–[31]. Additionally, we compared our system to recent transformer-based methods pre-trained on biomedical texts, namely BioBERT [34] and SciBERT [35]. We fine-tuned both BioBERT and SciBERT on each corpus, reporting the average performance using 10-fold cross validation. We used the official implementation and optimal hyper-parameters provided by the respective authors [34], [35].

Table 3 shows the performance of our system across corpora compared to other methods. Our system achieves the highest precision on all the corpora (93.2%, 90.7%, and 91.7% on LLL, HPRD50, and IEPA, respectively), outperforming by a large margin the BERT-based approaches in the precision metric while maintaining a F1 score comparable to other methods. The only exception is on the LLL corpus, where transformer-based methods and the "DSTK & feature kernel" approach achieve a very high F1 score. It is worth noting that our relation extraction approach also achieves the highest F1 score (81.1%) on the IEPA corpus, and differently from machine learning based systems, it does not need and rely on training data. Additionally, since our approach is a single system for all the corpora, it can be used as is on new data (see Section V), a typical requirement in industrial scenarios. These results strongly meet our expectations, since our goal was developing a high-precision system to allow researchers, and in particular biologists, to obtain reliable information without having to manually review the results and discard all the false positive instances. The relation extraction results on benchmark corpora can be further explored at: `https://apps.cosbi.eu/high-precision-nlp-benchmark/`.

A. ERROR ANALYSIS

To get additional insights on our approach, we analyzed both the false positives and the false negatives the system produces in order to make room for future work. A complete list of all the errors is provided in Supplementary File 2. We identified three sources of false positives, also summarized in Fig. 3:

- **Annotation inconsistencies.** Most false positive results (i.e., 58.62%) are caused by annotation inconsistencies in the corpora. We found sentences in which a relation, on a grammatical basis, actually exists, but it has not been annotated. For instance, in the following sentence[9]:

  *Several distinct mutations in exon2 of **VHL** disrupt binding of **pVHL** to TBP-1.*

  a relation between "VHL" and "pVHL" has not been annotated even if it is stated in the text. This could be due to the complex mutation statement that is described, which may be considered a biomedical event;

[9]Corpus: HPRD50, sentence ID: d26.

- **Dependency parsing errors.** The 20.69% of false positives is due to errors in the dependency parse tree. For instance, in the sentence[10]:

  *A low level of GerE **activated transcription** of cotD by sigmaK RNA polymerase in vitro, but a higher level of GerE repressed cotD transcription.*

  the verb "activated" has *amod* as the head relation label, denoting it is the adjectival modifier of "transcription";
- **Algorithm errors.** Other sources of errors account for the 20.69% of the total, and are mainly due to articulated syntactic structures that our algorithm wrongly navigates. For example, in the following sentence[11]:

  *These results clearly demonstrate that **UCP3** gene expression is upregulated by TZDs in the WAT and BAT in Wistar fatty rats, an obese model with **leptin** receptor defect, and that adipose UCP3 gene expression is increased in response to TZDs in vitro.*

  our system incorrectly identifies a relation between the biomedical entities "UCP3" and "leptin".

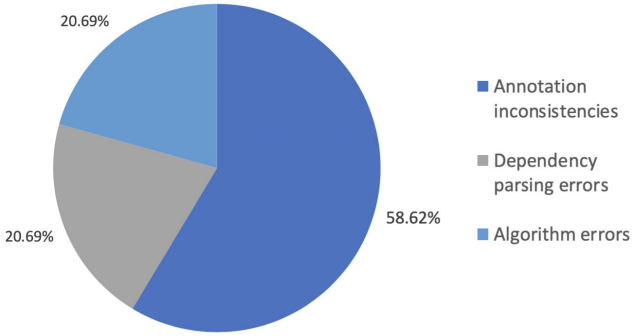


FIGURE 3. Distribution of the sources of false positive errors across corpora.

False positive errors can be instead classified depending on both the relation category they have been tested on, and their cause. Fig. 4 summarizes the distribution of false negatives according to this classification. Particularly, the 87.50% of false negatives belong to the R_V category, the 11.03% belong to the R_N category, and the 0.74% fall into the R_C category. The remaining 0.74% are cases that do not belong to any of the previous categories. For each category, the causes of false negatives we identified are the following:

- **Dependency parsing errors.** In the 64.71% of the total cases, false negatives are caused by errors in the dependency tree of the sentence being analyzed. For example, in the following sentence[12]:

  *We have shown previously that the transcription of degR is driven by an alternative sigma factor, **sigmaD**.*

[10]Corpus: LLL, sentence ID: d18.
[11]Corpus: IEPA, sentence ID: d88.
[12]Corpus: LLL, sentence ID: d26.

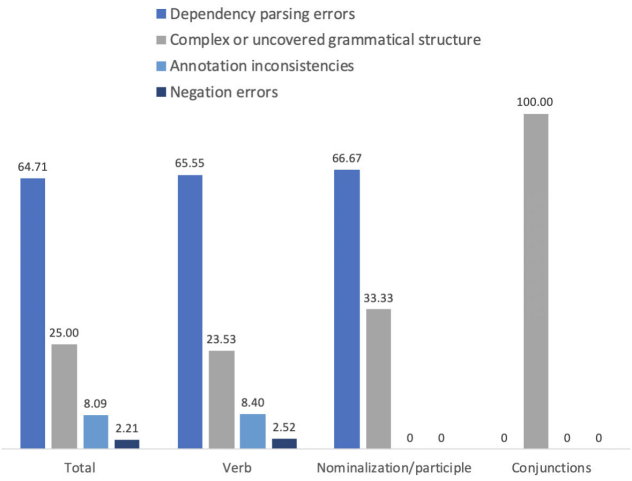


FIGURE 4. Distribution of the sources of false negative errors across corpora, according to both relation categories and their causes.

"sigmaD" is labeled as an appositional modifier (i.e., *appos*) of the verb "shown"; however, its head should instead be "factor". This results in a wrong structure that prevents our algorithm to correctly navigate the tree. We found this kind of error particularly prominent within the R_V category (i.e., 65.55%) and the R_N category (i.e., 66.67%). No errors of this kind are found in R_C;

- **Complex or unconvered grammatical structure.** In the 25.00% of the cases, the grammatical structure of the sentence has more than one subordinate or coordinate clause, and it is not easy to route. To give an example of this latter case, we can look at the following sentence[13]:

  ***SpoIIID** at low concentration repressed cotC transcription, whereas a higher concentration only partially repressed **cotX** transcription and had little effect on cotB transcription.*

  where to identify the actual relation between "SpoIIID" and "cotX", the system should be able to figure out that "higher concentration" is actually referring to "SpoIIID". However, this is far beyond the capabilities of our algorithm. While this false negative cause accounts for all the error within the R_C category, it only accounts for the 23.53% and the 33.33% within the R_V and R_N categories, respectively;
- **Annotation inconsistencies.** Similarly to the false positive analysis, false negatives could also be due to annotation inconsistencies. These errors account for the 8.09% of the total false negatives, and an example of this error type is exemplified by the following sentence[14]:

  *The aim of this study was to investigate the effects of hCG, hCG plus oxytocin and **oxytocin** on [3H] **inositol phosphate** (IP) formations in porcine*

[13]Corpus: LLL, sentence ID: d27.
[14]Corpus: IEPA, sentence ID: d17.

myometrial cells obtained from ovariectomized and cyclic gilts.

where "oxytocin" and "inositol phosphate", following the annotation standards of the corpora, are not actual relations, but instead statements about the purpose of the study. Fortunately, these errors are not common, representing only the 8.40% of the total errors in R_V;

- **Negation errors.** The remaining false negatives (i.e., 2.21%) are due to errors by our negation detector. For instance, in the sentence[15]:

  *From these results we conclude that **ComK** negatively regulates **degR** expression by preventing sigmaD-driven transcription of degR, possibly through interaction with the control region.*

  our system misses the relation between "ComK" and "degR". This is due to difficulties in discerning negated relations from negative relations. This error type is only present within the R_V category, accounting for a relative amount of 2.52% of the errors.

B. ABLATION STUDY

In order to provide additional insights on our method, we investigate the contribution of each rule category on the final performance of the system. In Table 4 we report precision, recall and F1 score on all corpora when R_V, R_N, R_C, and negation rule components are individually removed. As expected, the negation rules are crucial to the precision of the relation extraction system. In fact, when removed the precision score decreases on all the corpora (-3.1%, -8.3%, and -6.6% on LLL, HPRD50, and IEPA, respectively). We also notice a small increase in the F1 score on the LLL corpus (+2.0%). This is due to the characteristics of LLL, which exhibit few negated relations with respect to the other corpora. When removing R_V, R_N, and R_C, we obtain deeper insights about the importance of each relation category. For instance, the relation expressed by a verb (R_V) is by far the most important rule set. When removed, the precision increases on all the corpora (+6.8%, +3.7%, and +2.2% on LLL, HPRD50, and IEPA, respectively), while an important decrease appears evident in the recall metric (-58.0%, -58.4%, and -49.1% on LLL, HPRD50, and IEPA, respectively) and thus in the F1 score. This behaviour confirms that the R_V category is the primary source of errors of our system, but also the mean of a tradeoff between a very high precision and a satisfying recall. We notice a similar but less pronounced trend when removing relations expressed by nominalizations or participles (R_N). On the other hand, the category of relations expressed by conjunctions (R_C) contributes a little on all corpora. Particularly, it improves the precision (+0.1%), the recall (+0.7%), and the F1 score (+0.5%) on HPRD50, whereas it decreases the precision (-0.4%) and the F1 score (-0.2%) on IEPA.

[15]Corpus: LLL, sentence ID: d26.

TABLE 4. Ablation study on the contribution of each rule type. We report precision, recall, and F1 score of the relation extraction system on all the corpora when each rule category is removed.

Corpus	Configuration	P	R	F1
LLL	**Complete rule set**	93.2	73.1	81.9
	– R_V category rules	100.0	15.1	26.2
	– R_N category rules	91.5	58.1	71.1
	– R_C category rules	93.2	73.1	81.9
	– Negation rules	90.1	78.5	83.9
HPRD50	**Complete rule set**	90.7	70.8	79.5
	– R_V category rules	94.4	12.4	21.9
	– R_N category rules	90.6	63.5	74.7
	– R_C category rules	90.6	70.1	79.0
	– Negation rules	82.4	71.5	76.6
IEPA	**Complete rule set**	91.7	72.8	81.1
	– R_V category rules	93.9	23.7	37.9
	– R_N category rules	91.5	50.2	64.8
	– R_C category rules	92.1	72.8	81.3
	– Negation rules	85.1	73.2	78.7

C. LIMITATIONS AND OUTLOOK

Despite the good results, we identified some limitations which could be tackled in future work. Our system is able to extract highly precise binary relations, however there are use cases in which it would be useful to extract high-order associations (i.e., relations of relations), making a relation the argument of another relation, or modeling relations with more than two arguments. These requirements go beyond the purpose of this paper since we have focused on relation extraction and gold-standard annotations proposed in literature. We thus plan to enrich our system with this enhanced representation in future work, following the recent trends in event extraction [8]. Another limitation is about the algorithm errors, and in particular some difficult cases we presented in Section IV-A. We decided to rely on a rule-based method instead of using a machine learning approach to have a high degree of control on the behavior of the system, and to avoid to depend on application-specific training data. We designed rules as general as possible, relying only on syntactic information thus avoiding to overfit to words or corpus-specific constructs. This is a strong point in favour of our approach, since we are able to use the same system with the same rules across multiple corpora, obtaining high performance on all of them without retraining it on new target data. However, even if we employed a general approach, there are cases the system still does not capture, and where a machine learning system can be complementary. We thus plan to combine the complementary power of both rules and machine learning methods in future work. An interesting research direction is to exploit our flexible rule sets in a postprocessing stage to refine the results of a neural relation extraction method.

V. CASE STUDY

We present a case study on the mTOR signaling pathway [39] in order to show how our relation extraction system can be used in an industrial scenario. We have queried PubMed and

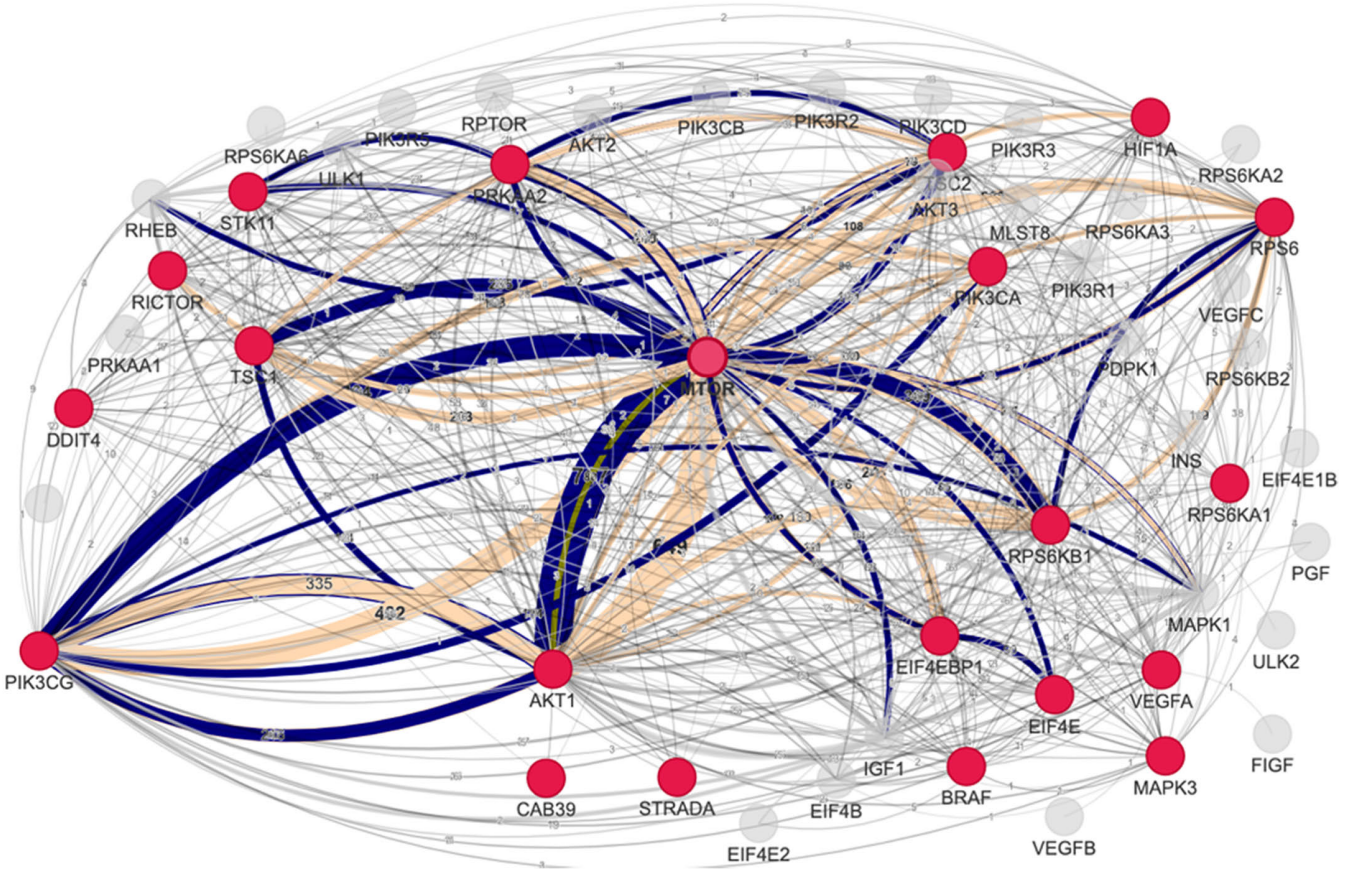


FIGURE 5. A view of the evidence relation network about the mTOR signaling pathway our relation extraction system produces. Nodes indicate proteins or genes, while edges represent semantic relationships between them. The network is filtered to only show relations with > 75 sentences supporting them. Orange edges: ***ASSOCIATED_WITH***; Blue edges: ***AFFECTS***; Brown edges: ***MEASURES***; Grey edges: all relations with ≤ 75 supporting sentences.

PMC to get all the relevant documents about the mTOR signaling pathway. The search has been performed ensuring the documents contain "mTOR pathway" in the title, abstract, MeSH (Medical Subject Headings) terms, or keywords, while asserting at least two proteins or genes belonging to the pathway – according to KEGG[16] (Kyoto Encyclopedia of Genes and Genomes) – are present in any position of the documents. A list of proteins/genes and their aliases is provided in Supplementary File 3. The query returned a total of 5,657 documents.

In order to find the semantic relationships between the actors of the pathway, we have firstly searched the proteins and genes within the documents using a dictionary-based approach. Those entities have been looked up using the Aho-Corasick algorithm [40] with their common textual variants: (i) hyphenation: search the entity also without hyphens; (ii) Greek symbols or words: search the entity also with the corresponding uppercase and lowercase Greek symbols ("α", "β", etc.) and words (e.g., "alpha", "beta", etc.), (iii) case: search the entity regardless of its letter case, and (iv) lemma: search the entity in its lemma form to abstract both the word person and the verb tense.

[16]https://www.genome.jp/kegg-bin/show_pathway?hsa04150

Then, our relation extraction system has been used to find relevant associations of those concepts, resulting in 22,379 evidence sentences from the literature. We have also assigned the relation (i.e., R_V, R_N, or R_C) a label indicating a semantic category by taking its lemma, and looking it up in a manually curated biomedical lexicon comprising 4,600 verbs in a lemma form together with their categories (e.g., *changed* → *AFFECTS*). This resource has been manually curated by field experts [41], and refined by biologists in our R&D team (Supplementary File 4).

Fig. 5 shows the resulting relation network, where nodes are the proteins and genes of the pathway, and edges represent evidence relations having more than 75 sentences supporting them (orange: *ASSOCIATED_WITH*, blue: *AFFECTS*, brown: *MEASURES*). It is worth noting that the knowledge base is not intended to be a biological "network pathway": a biomedical relation could in fact be stated even if two actors are interacting in the long run rather than directly. This is of particular interest to biologists, since the network is not restricted to show only evidence sentences about direct interactions. As a proof of concept, we hereafter present some associations identified by our relation extraction system:

- **Document PMID: 28086757**. The system retrieved the *ASSOCIATED_WITH* relations (*TSC1*, *involved*,

mTOR) and (*TSC2*, *involved*, *mTOR*) from the following sentence:

> *Rapamycin is used to treat tuberous sclerosis, a disease caused by mutations in either of the genes* ***TSC1*** *or* ***TSC2****, both of which are involved in the regulation of the* ***mTOR*** *pathway [13, 14].*

- **Document PMID: 29371951**. The system returned an *AFFECTS* relation (*BRAF*, *inhibit*, *AMPK*) from the following sentence:

  > *Oncogenic* ***BRAF*** *V600E mutant can inhibit the activity of* ***AMPK*** *by promoting phosphorylation of LKB1 and that this inhibition is critical for melanoma cell proliferation and growth.*

- **Document PMID: 29290965**. The system identified a *MEASURES* relation ($p-4E-BP1$, *predict*, *mTOR*) from the following sentence:

  > *Taken together, our and other studies suggest that* ***p-4E-BP1*** *may be an effective biomarker to predict* ***mTOR*** *inhibitor sensitivity in SCLC as well as in other cancers.*

VI. CONCLUSION

We presented a high-precision relation extraction system aiming to speed up the time-consuming process of the manual curation of semantic biomedical associations. Its rule-based design on syntactic dependency structures of texts gives the system the independence from specific training data, making it a *one-for-all* solution for industrial applications. Experimental results on gold-standard corpora showed that our method outperforms existing rule-based, feature- and kernel-based, and neural-based biomedical relation extraction approaches on the precision metric, while reaching a comparable or superior F1 score. Importantly, results indicated the high precision of our method is complementary to the high recall of transformer-based approaches, highlighting the need for more research on traditional linguistics-based methods. As a result, we met the requirement of limiting the expensive curation of the extracted semantic biomedical relationships to smoothly and reliably translate the extracted information into actionable knowledge. We plan to improve our methods by means of the richer representation of event extraction, exploiting the complementarity of both our rule sets and recent deep learning approaches, by blending them into a single system, one acting as a corrector of the other.

ACKNOWLEDGMENT

The authors would like to thank S. Michelini for the feedback on the work. *(Alan Ramponi and Stefano Giampiccolo contributed equally to this work.)*

REFERENCES

[1] R. A. Baasiri, S. R. Glasser, D. L. Steffen, and D. A. Wheeler, "The breast cancer gene database: A collaborative information resource," *Oncogene*, vol. 18, no. 56, pp. 7958–7965, Dec. 1999, doi: 10.1038/sj.onc.1203335.

[2] *PubMed: The Bibliographic Database, the NCBI Handbook*, 2nd ed., Nat. Center Biotechnol. Inf., Bethesda, MD, USA, 2013, pp. 13–24.

[3] *PubMed Central, the NCBI Handbook*, 2nd ed., Nat. Center Biotechnol. Inf., Bethesda, MD, USA, 2013, pp. 25–59.

[4] S. J. Nelson, "Medical terminologies that work: The example of MeSH," in *Proc. 10th Int. Symp. Pervas. Syst., Algorithms, Netw.*, Kaohsiung, Taiwan, 2009, pp. 380–384.

[5] Y. Tsuruoka, J. Tsujii, and S. Ananiadou, "FACTA: A text search engine for finding associated biomedical concepts," *Bioinformatics*, vol. 24, no. 21, pp. 2559–2560, Nov. 2008, doi: 10.1093/bioinformatics/btn469.

[6] D. Cheng, C. Knox, N. Young, P. Stothard, S. Damaraju, and D. S. Wishart, "PolySearch: A Web-based text mining system for extracting relationships between human diseases, genes, mutations, drugs and metabolites," *Nucleic Acids Res.*, vol. 36, no. 2, pp. W399–W405, May 2008, doi: 10.1093/nar/gkn296.

[7] Y. Liu, Y. Liang, and D. Wishart, "PolySearch2: A significantly improved text-mining system for discovering associations between human diseases, genes, drugs, metabolites, toxins and more," *Nucleic Acids Res.*, vol. 43, no. W1, pp. W535–W542, Jul. 2015, doi: 10.1093/nar/gkv383.

[8] S. Ananiadou, S. Pyysalo, J. Tsujii, and D. B. Kell, "Event extraction for systems biology by text mining the literature," *Trends Biotechnol.*, vol. 28, no. 7, pp. 381–390, Jul. 2010, doi: 10.1016/j.tibtech.2010.04.005.

[9] C.-C. Huang and Z. Lu, "Community challenges in biomedical text mining over 10 years: Success, failure and the future," *Briefings Bioinf.*, vol. 17, no. 1, pp. 132–144, Jan. 2016, doi: 10.1093/bib/bbv024.

[10] D. Szklarczyk, J. H. Morris, H. Cook, M. Kuhn, S. Wyder, M. Simonovic, A. Santos, N. T. Doncheva, A. Roth, P. Bork, L. J. Jensen, and C. von Mering, "The STRING database in 2017: Quality-controlled protein–protein association networks, made broadly accessible," *Nucleic Acids Res.*, vol. 45, no. D1, pp. D362–D368, Jan. 2017, doi: 10.1093/nar/gkw937.

[11] O. V. Saik, T. V. Ivanisenko, P. S. Demenkov, and V. A. Ivanisenko, "Interactome of the hepatitis c virus: Literature mining with ANDSystem," *Virus Res.*, vol. 218, pp. 40–48, Jun. 2016, doi: 10.1016/j.virusres.2015.12.003.

[12] J. Zhou and B.-Q. Fu, "The research on gene-disease association based on text-mining of PubMed," *BMC Bioinf.*, vol. 19, no. 1, pp. 1–8, Dec. 2018, doi: 10.1186/s12859-018-2048-y.

[13] B. Bhasuran and J. Natarajan, "Automatic extraction of gene-disease associations from literature using joint ensemble learning," *PLoS ONE*, vol. 13, no. 7, Jul. 2018, Art. no. e0200699, doi: 10.1371/journal.pone.0200699.

[14] J. Pinero, N. Queralt-Rosinach, A. Bravo, J. Deu-Pons, A. Bauer-Mehren, M. Baron, F. Sanz, and L. I. Furlong, "DisGeNET: A discovery platform for the dynamical exploration of human diseases and their genes," *Database*, vol. 2015, Apr. 2015, Art. no. bav028, doi: 10.1093/database/bav028.

[15] M. Bundschus, M. Dejori, M. Stetter, V. Tresp, and H.-P. Kriegel, "Extraction of semantic biomedical relations from text using conditional random fields," *BMC Bioinf.*, vol. 9, no. 1, p. 207, 2008, doi: 10.1186/1471-2105-9-207.

[16] A. P Tafti, J. Badger, E. LaRose, E. Shirzadi, A. Mahnke, J. Mayer, Z. Ye, D. Page, and P. Peissig, "Adverse drug event discovery using biomedical literature: A big data neural network adventure," *JMIR Med. Informat.*, vol. 5, no. 4, p. e51, Dec. 2017, doi: 10.2196/medinform.9170.

[17] A. B. Abacha, M. F. M. Chowdhury, A. Karanasiou, Y. Mrabet, A. Lavelli, and P. Zweigenbaum, "Text mining for pharmacovigilance: Using machine learning for drug name recognition and drug–drug interaction extraction and classification," *J. Biomed. Informat.*, vol. 58, pp. 122–132, Dec. 2015, doi: 10.1016/j.jbi.2015.09.015.

[18] J. X. Binder, S. Pletscher-Frankild, K. Tsafou, C. Stolte, S. I. O'Donoghue, R. Schneider, and L. J. Jensen, "COMPARTMENTS: Unification and visualization of protein subcellular localization evidence," *Database*, vol. 2014, Feb. 2014, Art. no. bau012, doi: 10.1093/database/bau012.

[19] K. Fundel, R. Kuffner, and R. Zimmer, "RelEx—Relation extraction using dependency parse trees," *Bioinformatics*, vol. 23, no. 3, pp. 365–371, Feb. 2007, doi: 10.1093/bioinformatics/btl616.

[20] K. Yu, P.-Y. Lung, T. Zhao, P. Zhao, Y.-Y. Tseng, and J. Zhang, "Automatic extraction of protein-protein interactions using grammatical relationship graph," *BMC Med. Informat. Decis. Making*, vol. 18, no. S2, p. 42, Jul. 2018, doi: 10.1186/s12911-018-0628-4.

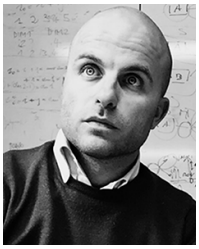
[21] T. T. T. Phan and T. Ohkawa, "Protein-protein interaction extraction with feature selection by evaluating contribution levels of groups consisting of related features," *BMC Bioinf.*, vol. 17, no. S7, p. 246, Jul. 2016, doi: 10.1186/s12859-016-1100-z.

[22] S. Kim, J. Yoon, J. Yang, and S. Park, "Walk-weighted subsequence kernels for protein-protein interaction extraction," *BMC Bioinf.*, vol. 11, no. 1, p. 107, Dec. 2010, doi: 10.1186/1471-2105-11-107.

[23] M. Miwa, R. Sætre, Y. Miyao, and J. Tsujii, "Protein–protein interaction extraction by leveraging multiple kernels and parsers," *Int. J. Med. Informat.*, vol. 78, no. 12, pp. e39–e46, Dec. 2009, doi: 10.1016/j.ijmedinf.2009.04.010.
[24] N. Warikoo, Y.-C. Chang, and W.-L. Hsu, "LPTK: A linguistic pattern-aware dependency tree kernel approach for the BioCreative VI CHEMPROT task," *Database*, vol. 2018, Jan. 2018, Art. no. bay108, doi: 10.1093/database/bay108.
[25] Y.-C. Chang, C.-H. Chu, Y.-C. Su, C. C. Chen, and W.-L. Hsu, "PIPE: A protein–protein interaction passage extraction module for BioCreative challenge," *Database*, vol. 2016, Aug. 2016, Art. no. baw101, doi: 10.1093/database/baw101.
[26] G. Murugesan, S. Abdulkadhar, and J. Natarajan, "Distributed smoothed tree kernel for protein-protein interaction extraction from the biomedical literature," *PLoS ONE*, vol. 12, no. 11, Nov. 2017, Art. no. e0187379, doi: 10.1371/journal.pone.0187379.
[27] Z. Zhao, Z. Yang, H. Lin, J. Wang, and S. Gao, "A protein-protein interaction extraction approach based on deep neural network," *Int. J. Data Mining Bioinf.*, vol. 15, no. 2, pp. 145–164, 2016, doi: 10.1504/IJDMB.2016.076534.
[28] Y. Zhang, H. Lin, Z. Yang, J. Wang, S. Zhang, Y. Sun, and L. Yang, "A hybrid model based on neural networks for biomedical relation extraction," *J. Biomed. Informat.*, vol. 81, pp. 83–92, May 2018, doi: 10.1016/j.jbi.2018.03.011.
[29] S. Yadav, A. Ekbal, S. Saha, A. Kumar, and P. Bhattacharyya, "Feature assisted stacked attentive shortest dependency path based bi-LSTM model for protein–protein interaction," *Knowl.-Based Syst.*, vol. 166, pp. 18–29, Feb. 2019, doi: 10.1016/j.knosys.2018.11.020.
[30] H. Zhang, R. Guan, F. Zhou, Y. Liang, Z.-H. Zhan, L. Huang, and X. Feng, "Deep residual convolutional neural network for protein-protein interaction extraction," *IEEE Access*, vol. 7, pp. 89354–89365, 2019, doi: 10.1109/ACCESS.2019.2927253.
[31] M. Ahmed, J. Islam, M. R. Samee, and R. E. Mercer, "Identifying protein-protein interaction using tree LSTM and structured attention," in *Proc. IEEE 13th Int. Conf. Semantic Comput. (ICSC)*, Newport Beach, CA, USA, Jan. 2019, pp. 224–231.
[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, Long Beach, CA, USA, 2017, pp. 5998–6008.
[33] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol. (NAACL-HLT)*, Minneapolis, MN, USA, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
[34] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "BioBERT: A pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, Sep. 2019, doi: 10.1093/bioinformatics/btz682.
[35] I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A pretrained language model for scientific text," in *Proc. Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, Hong Kong, 2019, pp. 3615–3620, doi: 10.18653/v1/D19-1371.
[36] C. Nédellec, "Learning language in logic—Genic interaction extraction challenge," in *Proc. 4th Learn. Lang. Logic Workshop (LLL ICML)*, Bonn, Germany, 2005, pp. 1–81.
[37] J. Ding, D. Berleant, D. Nettleton, and E. Wurtele, "Mining MEDLINE: Abstracts, sentences, or phrases?" in *Proc. Pacific Symp. Biocomput. (PSB)*, vol. 7, 2002, pp. 326–337.
[38] S. Peri *et al.*, "Human protein reference database as a discovery resource for proteomics," *Nucleic Acids Res.*, vol. 32, no. 1, pp. 497D–501D, Jan. 2004, doi: 10.1093/nar/gkh070.
[39] M. Laplante and D. M. Sabatini, "mTOR signaling in growth control and disease," *Cell*, vol. 149, no. 2, pp. 274–293, Apr. 2012, doi: 10.1016/j.cell.2012.03.017.
[40] A. V. Aho and M. J. Corasick, "Efficient string matching: An aid to bibliographic search," *Commun. ACM*, vol. 18, no. 6, pp. 333–340, Jun. 1975, doi: 10.1145/360825.360855.
[41] Y. Kim, S. Beak, and M. Song, "Constructing linguistic verb source for relation extraction," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, Indianapolis, IN, USA, 2016, pp. 2511–2512.

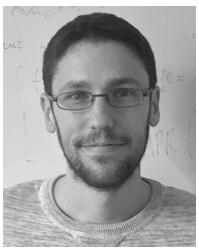
ALAN RAMPONI was born in Vicenza, Italy. He received the B.S. and M.S. degrees (Hons.) in computer science from the University of Trento, Italy, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree in computer science.

He is holding a fellowship from Fondazione The Microsoft Research—University of Trento Centre for Computational and Systems Biology, Rovereto, Italy. He has industry experience in the digital and innovation sector, and from 2016 to 2019, he was a Teaching Assistant at both the Department of Information Engineering and Computer Science, and the Department of Civil, Environmental and Mechanical Engineering, University of Trento. From 2019 to 2020, he was a Visiting Ph.D. Fellow with the NLP research group at the IT University of Copenhagen, Denmark. His research interests include natural language processing, transfer learning, and robustness in machine learning.



STEFANO GIAMPICCOLO was born in Trento, Italy, in 1986. He received the B.S. degree (Hons.) in mathematics from the University of Trento, Italy, in 2008, the M.S. degree (Hons.) in mathematics from the University of Pisa, Italy, in 2011, and the master's degree in innovation management from Scuola Superiore Sant'Anna, Pisa, Italy, in 2012.

From 2012 to 2016, he was with the Research and Development Department of a GIS company in Trento, Italy, where he worked on software development and computational geometry algorithms. Since 2017, he has been working with the Fondazione The Microsoft Research—University of Trento Centre for Computational and Systems Biology, Rovereto, Italy, focusing on the development of text-mining algorithms, visual models, and tools for data analysis and visualization.



DANILO TOMASONI received the B.S. and M.S. degrees in computer science from the University of Trento, Italy, in 2009 and 2011, respectively.

From 2008 to 2010, he was a Researcher and a Developer with the Embedded System Unit at FBK (Fondazione Bruno Kessler), Trento, Italy, in the field of hardware formal verification. From 2012 to 2014, he was a Researcher and a Developer with the Research and Development unit of a simulation based engineering company in the field of artificial intelligence. Since 2016, he has been a Researcher in the field of systems biology, data analysis, and modeling and simulation of biological systems at Fondazione The Microsoft Research—University of Trento Centre for Computational and Systems Biology, Rovereto, Italy. His research interests include artificial intelligence applied to natural language processing, systems biology, model checking, and cyber security. He also holds the CEH (Certified Ethical Hacker) certification from EC-Council.

CORRADO PRIAMI received the M.Sc. and Ph.D. degree in computer science from the University of Pisa.

He held a postdoctoral position with a competitive EU Marie Curie Grant at the Ecole Normale Superiéure, Paris, from 1996 to 1997, a Researcher and an Associate Professor with the University of Verona, from 1997 to 2001, a Visiting Scholar with Microsoft Corporation, in 2004, a Visiting Professor at Stanford University, from 2016 to 2017, and a Professor with the University of Trento, from 2001 to 2017. He is currently a Professor of computer science with the University of Pisa, the Director of the Pisa node of the Stanford SPARK Global initiative, and has more than 20 years of academic and industrial experience in the application of computational technology for pharma and food companies. The results of his Ph.D. thesis on stochastic pi-calculus were the basis for the foundation of COSBI, that he led more than 12 years as the Founder, President, and CEO. He is currently the Founder and CSO of COSBI. In early 2018, he also joined Vydiant, a California-based, health tech company, as CTO. He served in the Senate of the University of Verona, in the BoD of the University of Trento, in the BoD of the Trento School of Management, and in the BoD of COSBI as Chairman of the board. He published over 200 scientific articles, gave more than 100 invited talks and lectures, regularly serves in advisory and scientific boards (including the Stanford SPARK program) as well as in reviewing panels for international funding agencies and institutions. He teached more than 2000 hours in the fields of programming languages and bioinformatics at undergraduate and graduate level. He supervised more than 100 people (students, Ph.D. students, and Post-Docs) of which about 40 are now in senior or research positions in academia and industry.

ROSARIO LOMBARDO was born in Phoenix, AZ, USA. He received the B.Sc. and M.Sc. degrees *(*both *cum laude)* in computer science from the University of Pisa, the M.B.A. degree in consulting management from the SP Jain School of Global Management (Dubai, Sydney, Singapore), and the Ph.D. degree in bioinformatics from the University of Verona.

He is currently the Head of the Bioinformatics, Fondazione The Microsoft Research—University of Trento, where he has been supervising several research projects in collaboration with pharma and nutrition companies, driving the innovation towards industrial scientific research. Among his research interests are deep learning and text mining, enabling technologies for complex quantitative systems pharmacology models such as visual modeling, high-performance simulations. He was a Lecturer with the Universities of Pisa and Trento and has been mentoring several intern/B.Sc./M.Sc./Ph.D. students. In over 15 years' experience in business, scientific, and technological consulting, he was coach to a number of colleagues and has managed cross-functional, international projects in Pharma, Nutrition, Academia and Banking. He has entrepreneurial experience in tech-enabled ventures.