Interdisciplinary Information Sciences Vol. 22, No. 1 (2016) 87–100

ISSN 1340-9050 print/1347-6157 online
DOI 10.4036/iis.2016.R.03

A Hybrid Method for Open Information Extraction Based on Shallow and Deep Linguistic Analysis

Vahideh RESHADAT[1,*], Maryam HOORALI[1] and Heshaam FAILI[2]

[1]*Faculty of Information and Communication Technology, Malek-Ashtar University of Technology, Tehran, Iran*
[2]*School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran*

Open Information Extraction is a relation-independent extraction paradigm that extracts assertions from massive and heterogeneous corpora such as the Web. Light relation extractors focus on efficiency by restricting analysis to some shallow linguistic tools such as part-of-speech tagging. Although these methods are fast and scalable, they are unable to deal with complex sentences (such as complicated and long distance relations) due to using only shallow syntactic features. This paper presents two novel hybrid methods, TextRunner-DepOE (TR-DOE) and ReVerb-DepOE (RV-DOE) which combine high-performance subset of shallow Open IE systems with the strengths of a deep Open IE system. We detect the best trade-off between precision and recall by tuning two combination parameters: sentence length and confidence measure. Since the focus is on using time efficiently, we used a fast and robust deep extractor. Experiments indicate that the proposed hybrid methods obtain significantly higher performance than their constituent systems. The best result was for TR-DOE which had an F-measure almost twice that of TextRunner.

KEYWORDS: Information Extraction, Open Information Extraction, Relation Extraction, Knowledge Discovery, Fact Extraction

1. Introduction

There exists a vast amount of human knowledge residing in books, text documents and electronic texts on the web including newswire, blogs, email, and so on. However the largeness and heterogeneity of data make human annotation hard. A common way to help human to understand these data is to transfer unstructured text into structured representation. Turning unstructured text into machine-readable knowledge requires transforming syntactic construct to semantic representation [1].

Information Extraction is the task of automatically processing unstructured or semi-structured text and generating a database of entities, relations and events [2, 3]. One of the core information extraction tasks is relation extraction which aims at extracting semantic relations among entities from natural language text. Relation extraction can potentially benefit a wide range of NLP tasks such as Web search, question answering, ontology learning, summarization, building knowledge bases etc. [4, 5]. Due to these many potential applications, relation extraction has gained much attention in information extraction.

The huge and fast-growing scale, a mixed genre of documents and infinite types of relations are challenges of the Web-scale relation extraction [6]. Therefore, there is a need to a scalable system to extract information from large heterogeneous Web-scale open domain. The traditional approaches to information extraction (such as [7–9]) assume a fixed set of predefined target relations and usually do not scale to corpora where the number of target relations is very large [10]. An alternative paradigm, Open Information Extraction (Open IE) aims to scale information extraction methods to the size and diversity of the Web corpus. Open IE systems extract relational tuples from text, without requiring a pre-specified vocabulary [11, 12]. The key goals of Open IE are: (1) domain independence, (2) unsupervised extraction, and (3) scalability to large amounts of text [13].

Some approaches such as TextRunner [14], WOE_{pos} [15], ReVerb [16], R2A2 [17] and the method presented in [18] focus on efficiency by restricting syntactic analysis to part-of-speech tagging and chunking. These extractors are usually fast but the restriction to shallow syntactic analysis limit maximum recall and/or may lead to a significant drop of precision at higher points of recall.

Other approaches such as Wanderlust [19], WOE_{parse} [15], KrakeN [20], OLLIE [11], ZORE [21], DepOE [22], SRL-IE-Lund [23], SRL-IE-UIUC [23], the methods proposed in [24] and [25] additionally use deep syntactic and semantic analysis such as dependency parsing. These extractors are generally more expensive than the previous extractors; they trade efficiency for improved precision and recall. The former extractors are rapid, guarantee scalability and require less effort due to using shallow syntactic features, while the latter extractors are efficient for

Received October 24, 2015; Accepted August 30, 2016
*Corresponding author. E-mail: vreshadat@mut.ac.ir

precision and recall but time consuming and require considerable effort due to using deep syntactic features in the extraction process.

Regarding to these problems, the current research seeks to evaluate the feasibility of designing and implementing a hybrid method based on shallow and deep Open IE extractors by using high performance subsets of them, so that the resulting system would be high performance and satisfy Open IE systems' goal. The huge and fast growing scale of the Web is one of the main challenges of the Open IE systems. By deploying deep extractors on the appropriate portion of the input data, hybrid methods can process more data in less time in comparison with the pure deep methods and hence, the speed of extraction is increased. One of the results of combining a shallow Open IE extractor with a deep one is a reduction in the defects of shallow Open IE system and, thus, an enrichment in its extractions, so that the resulting system guarantees scalability and achieves better performance. In this paper, we study the applicability of enriching shallow syntactic Open IE systems' extractions by using a deep syntactic Open IE approach. For this purpose, high performance subset of shallow Open IE methods' outputs was examined by using two parameters: sentence length and confidence measure. Sentences of input yielding low performance, are passed to the deep Open IE system. We detect the best trade-off between precision and recall by tuning combination parameters. For this purpose, different threshold values of confidence and sentence length are applied and the impact of these factors on resulting performance is assessed. Similar hybrid techniques have also been employed in other fields such as parsing (e.g., in [26]), where an architecture for integration of shallow and deep NLP components have been presented. In current work, since our goal is to improve performance in a fast way, DepOE [22] system is used. DepOE is a deep extractor that is faster in comparison with other systems such as Open IE systems that are based on semantic role labeling or deep syntactic parser. Sentence-based hybrid methods cause increasing of accuracy in the shallow extractors in less (or at most equal) time than taken by pure deep extractors. On the other hand, they increase speed and performance of deep extractors. These cases become important in large inputs such as Web-scale data. Confidence-based hybrid methods can also improve precision. Since they have time overhead, the employed deep method is fast enough in order to decrease total time as far as possible.

DepOE is one of the recent robust and fast, deep Open IE systems which uses dependency parsing. Since it is based on deep syntactic analysis, it achieves good level of precision. We first examine the applicability of using this method for improving performance of TextRunner [14] system. TextRunner is the first approach for Open IE which has achieved high performance in handling an unbounded number of relations. TextRunner uses shallow linguistic features and runs quickly enough to process Web-scale corpora. This hybrid method is called TextRunner-DepOE (TR-DOE).

Next we examine ReVerb [17], a successful shallow Open IE system which has attempted to improve precision by restricting relations to specific part-of-speech tag sequences that are presumed to express true relations [22]. It is believed that deep extractors in comparison to shallow extractors may result in a much higher precision and recall and also reduce the defects made by shallow linguistic analysis. We study the trade-off between ReVerb and DepOE across two metrics and examine the effect of metrics on efficiency of the resulted hybid system. This hybrid method is called ReVerb-DepOE (RV-DOE). The experiment results show that using deep extractors on the smart subset of shallow extractors' input sentences can yield a substantial improvement in F-measure.

In this paper, we address several empirical questions. Can a hybrid method based on shallow and deep Open IE extractors achieve high performance on the heterogeneous Web text? What is the relative trade-off between the two extractors? Does considered factors influence the relative performance of resulted hybrid methods? Finally, is it possible to design an efficient hybrid extractor to get the best of both deep and shallow extractors? This paper examines these questions and makes the following contributions:

- We present two novel hybrid approaches for Open IE from text using deep and shallow features, which exploit strengths of a deep Open IE system to enrich a shallow Open IE system's extractions in order to achieve better performance. Since a fast deep Open IE system is used in the structure of hybrid extractors, they are considered as fast hybrid methods.
- High performance subset of shallow Open IE methods' outputs was gained through two combination parameters: sentence length and confidence measure. Experiment results demonstrate that these parameters can help hybrid extractors to achieve better F-measure in comparison with their constituent systems.
- The main idea of sentence-based hybrid methods is reducing time and increasing performance (e.g., precision). In the confidence-based hybrid methods the main goal is also improving precision and as a result performance. Based on these goals hybrid methods were designed and they could satisfy desired goals.
- By keeping shallow Open IE extractions with high confidence and applying deep extractors for sentences yielding low confidence, proposed hybrid systems which are obtained through confidence parameter are able to achieve high performance. We also found that the best time efficiency for proposed hybrid systems were obtained through sentence length parameter, achieved using shallow extractors for short sentences and deep extractors for long sentences.

The rest of this paper is organized as follows. Section 2 introduces previous works in the areas of Open IE systems. Our proposed extractors are described in Sect. 3. We present results of our experiments in Sect. 4 and end with conclusion in Sect. 5.

2. Related Works

In this section we review some representative work on open information extraction, in particular works on open relation extraction. Open IE has received considerable attention in the last few years. Open IE systems can be divided into two categories based on the linguistic analysis which can be performed for relation extraction. Some Open IE approaches use only shallow analysis tools and other approaches additionally use deep analysis tools.

TextRunner [14], is first and high scalable Open IE system where the tuples are assigned a probability. Its extraction process contains three steps. (1) It trains a conditional random field based classifier on sample of triples generated from applying heuristics on a sample of 1,000 sentences of Penn Treebank for extracting relationships. The self-supervised nature of learner reduces the need for hand-labeled training data. TextRunner uses shallow features that are fast and easy to compute at extraction time. Deploying unlexicalized features in classifier makes it practical to perform at web-scale. (2) Extractor makes a single pass over the corpus and tries to generate as many candidate tuples as possible from sentences. The candidate instances are examined by classifier and trustworthy labelled ones are retained. (3) A redundancy based assessor examines each retained triple and rank them based on a probabilistic model of redundancy in text. TextRunner actually exploits a confidence measure which counts the number of distinct sentences from which each extraction was found. Assessor applies this count to determine a probability model and assigns a confidence score to each tuple. It assigns higher confidence to triples that have high number of occurring times. This confidence measure shows that a tuple is a correct instance of a relation among arguments. Extractions with low confidence are ignored by extractor.

ReVerb [16] is a strong and successful shallow Open IE system. It makes use of a simple part-of-speech tag sequence as a syntactic constraint in order to extract relation phrases and eliminate incoherent extractions and also reduce uninformative extractions. There are relation phrases that satisfy the syntactic constrain but are over-specified. To overcome this limitation, ReVerb exploits a lexical constraint that aims to alleviate the amount of over-specified extractions. This constrain is based on the intuition that a valid relation phrase should take many distinct arguments in a large corpora. ReVerb uses a logistic regression classifier to determine a confidence score for each triple. All features used by the classifier are relation independent and fast computed. Experiments show ReVerb outperforms TextRunner and its performance is more than twice that of TextRunner [15, 27].

WOE_{pos} [15] is limited to shallow features like part-of-speech tags and uses Wikipedia for self-supervised learning of unlexicalized extractor. In fact, it autonomously transfers knowledge from random editors' effort of collaboratively editing Wikipedia to train an open information extractor. Specifically, WOE_{pos} generates relation-specific training examples by matching Infobox attribute values to corresponding sentences and abstracts these examples to relation-independent training data to learn an unlexicalized extractor. R2A2 [16] uses an argument learning component. It makes use of a number of classifiers to identify the arguments of a verbal phrase (based on hand-labeled training data). Two classifiers identify the left and right bounds for first argument and one classifier identifies the right bound of second argument. R2A2 is able to extract propositions that contain arguments that are not noun phrases.

Wanderlust [19] uses hand-labeled training data to learn extraction patterns on the dependency tree. After annotating 10,000 sentences parsed with LinkGrammar, it learns 46 general linkpaths as patterns for relation extractions. WOE_{parse} [15] is a pattern classifier learned from dependency path patterns which uses typed dependencies as features. WOE_{parse} learned more than 15,000 patterns over typed dependencies. Authors in their evaluation showed that using deep syntactic parsing improves the precision of their system, however at a high cost in extraction speed.

ZORE [21] is a syntax-based Chinese open relation extraction system for extracting relations and semantic patterns from Chinese text. ZORE identifies relation candidates from automatically parsed dependency trees, and then extracts relations with their semantic patterns iteratively through a double propagation algorithm. [25] Also focuses on Chinese Open relation extraction. This system can be considered as a pipeline of word segmentation, parts of speech (POS) tagging and parsing.

Dependency-based Open information Extraction (DepOE) [22] is a system based on fast dependency parsing which has the important feature of being able to operate at Web-scale. This method is organized in three main steps. (1) Dependency parsing: It uses DepPattern, a dependency-based parser which is a multilingual tool, to analyse each sentence of the input text. (2) Clause constituents: for each parsed sentence, the verb clauses are detected and then for each clause, the verb participants are identified. (3) A small set of extraction rules are applied on the discovered clauses constituents in order to extract the target triples.

LSOE [28] is an Open IE extractor based on lexical-syntactic patterns. The main idea is to provide a simple solution to perform rule-based extraction of triples using POS-tagged text. The method is developed based on two kinds of patterns: (1) generic patterns (2) rules from Cimiano and Wenderoth proposal [29]. LSOE performance was compared with ReVerb and DepOE. The results show that LSOE extracts relations that are not learned by other extractors and achieves compatible precision.

Most Open IE approaches are designed to capture binary facts and they usually do not extract higher order N-ary facts. KrakeN [20] addresses this shortage. It can extract more facts per sentence in high precision and is capable of capturing unary, binary and higher order N-ary facts. Since using a dependency parser results in cost in speed and

recall, many sentences were ignored due to heuristic of detecting erroneous parses. OLLIE [11] aims to improve the Open IE systems by using a hybrid approach based on bootstrapping. It learns pattern templates automatically from a training set that is bootstrapped from relations extracted by the ReVerb system. It obtains the pattern templates from the dependency path connecting pairs of entities and their corresponding relations. The patterns are then applied over the corpus and new facts are obtained. OLLIE produces n-ary extractions by merging binary relations. It has 1.9 to 2.7 times more area under precision-yield curves compared to ReVerb and WOE. ClauseIE [13] is a novel, clause-based approach to open information extraction which differs from previous approaches in that it separates the detection of "useful" pieces of information expressed in the sentence from their representation in terms of extractions. ClauseIE uses linguistic knowledge about the grammar of the English language to first detect clauses in an input sentences and to subsequently identify the type of each clause according to the grammatical function of its constituents. ClauseIE is based on dependency parsing and a small set of domain-independent lexica. It examines sentences without any post-processing and requires no training data. ClauseIE achieves high precision and recall and can be customized to output triples or n-ary facts. EXEMPLAR [30] is an open relation extraction method that extracts n-ary relations. It uses rules over dependency parse trees to detect relation instances. EXEMPLAR's rules are applied to each candidate argument individually as opposed to all candidate arguments of an instance. Since the goal is to achieve high precision and low computational cost, EXEMPLAR variations have been presented by different dependency parsers. The results are promising and EXEMPLAR outperforms the systems that support n-ary extraction.

Bast and Haussman [31] proposed a method called CSD-IE that uses contextual sentence decomposition for Open IE. It decomposes a sentence into the parts that semantically 'belong together'. The facts are then obtained by identifying the (implicit or explicit) verb in each part. This system has been compared to ReVerb, OLLIE and ClauseIE. This comparison considers accuracy, minimality, coverage and number of extracted facts. Minimality examines if the extracted triple can be further decomposed into smaller meaningful triples. Percentage of text contained in at least one extracted triple is called coverage. CSD-IE outperforms ReVerb and OLLIE in terms of coverage and recall, but at comparable accuracy and minimality. It achieves precision and recall comparable to ClausIE, but at significantly better minimality. In [32], the same authors improved the informativeness of extracted facts in Open IE by using some inference rules. Uninformative extracted facts are problem for semantic search applications utilizing them. Their evaluation shows that this approach can increase the number of correct and informative triples by 15% discarding the uninformative ones. The notion of informativeness used in their evaluations is stricter than the one used by Fader et al. [16].

The TreeKernel [33] method considers two subtasks for open relation extraction: first examining whether there is a relation between two name entities in a sentence, and then whether there are explicit relation words for two entities. Several SVM models with dependency tree kernels have been leveraged. TreeKernel generates candidate relations by leveraging a set of syntactic patterns. Then candidate instances are produced by incorporating all possible entity pairs with extracted relations from a sentence. Given a candidate instance, TreeKernel extracts some paths by using dependency tree. A tree kernel classifier then identifies a correct relation instance. Although this system is superior to some systems like ReVerb and OLLIE in terms of the both subtasks, it requires more computational time.

In [23] authors proposed an Open IE system based on semantic role labeling (SRL). They constructed novel extractors based on two semantic role labeling systems, one developed at UIUC's publicly available SRL system [34] and the other at LUND [35]. SRL-based open extractors perform computationally expensive operations but they are robust to noisy heterogeneous Web text and their deep processing results in high performance. Since these extractors required 2.5 orders of magnitude longer processing time than TextRunner, authors proposed hybrid methods that output higher quality results than TextRunner and similar quality as these extractors in much less time. RECALLHYBRID is a hybrid method which runs TextRunner over all the sentences and uses the remaining time to run SRL-IE on a random subset of the sentences and takes union of the extractions. PRECHYBRID is another hybrid method that focuses on increasing the precision and uses the filtering policy and intelligent order of sentences for extraction.

The tradeoffs between Open IE and traditional IE were investigated in [10]. According to this paper, when the number of relations is massive, and the relations themselves are not pre-specified, Open IE is necessary. Authors presented a new model for Open IE called O-CRF and showed that it achieves increased precision and nearly double the recall than the model employed by TextRunner. They also showed that when the number of target relations is small, and their names are known in advance, O-CRF is able to match the precision of a traditional extraction system, though at substantially lower recall. They combined the two types of systems into a hybrid that achieves higher precision than a traditional extractor, with comparable recall.

Existing Open Information Extraction systems have mainly focused on Web's heterogeneity rather than the Web's informality. The performance of the ReVerb system, drops dramatically as informality increases in Web documents. In [24] a Hybrid Ripple-Down Rules based Open Information Extraction (Hybrid RDROIE) system was proposed, which uses RDR on top of a conventional Open IE system. The Hybrid RDROIE system applies RDR's incremental learning technique as an add-on to the state-of-the-art ReVerb Open IE system to correct the performance degradation of ReVerb due to the Web's informality in a domain of interest. The Hybrid RDROIE system doubled ReVerb's performance in a domain of interest after two hours training.

Some related works to open relation extraction systems are semantic best-effort information extraction approaches. IE approaches can be complemented by alternative information extraction scenarios where accuracy, consistency and regularity are traded by domain-independency, context capture, wider extraction scope and maximization of the text semantics representation, under a pay-as-you-go data quality perspective [36], where data semantics and data quality are built and improved over time. An information extraction strategy which focuses on these aspects is called a semantic best-effort information extraction approach. In [37] an approach was proposed, which focuses on the analysis of a semantic best-effort extraction approach using structured discourse graphs (SDGs), a RDF-based discourse representation format. A semantic best-effort extraction pipeline was proposed and was implemented on the Graphia framework. The quality of the proposed extraction approach was evaluated over Wikipedia.

Semantic network is another related area. Graph-based representations of knowledge have been shown to provide both intuitive and formally rigorous access to the represented information. ASKnet system [38] uses NLP tools to extract semantic information from text, and then, through a novel use of spreading activation theory, combines that information into an integrated large-scale semantic network. By mapping together concepts and objects that relate to the same real-world entities, ASKNet is able to produce a single unified entity relationship style semantic network.

3. Proposed Hybrid Approach for Open Information Extraction

A large number of Open Information Extraction systems have been propose recently, covering a wide range of NLP tools, from shallow (e.g., part-of-speech tagging) to deep (e.g., semantic role labeling). Since the focus of Open IE is finding all the potentially useful facts from a huge and heterogeneous corpora such as Web in reasonable time, approaches that exploit heavier NLP analysis (syntactic or semantic tools such as semantic role labeling, parser, and so on) are costly. In addition, tools for automatic deep analysis are available only for a limited number of natural languages, and produce imperfect results. Manual deep analysis, on the other hand, is time consuming and expensive [39]. Another way to extract relations is to rely only on a shallow linguistic analysis by taking into account the lemmas, chunks or POS tags. Automatic tools for shallow analysis are available for many languages, and are sufficiently reliable [39]. These extractors are usually fast but the restriction to shallow syntactic analysis limits maximum recall and/or may lead to a significant drop of precision at higher points of recall [13]. Each of these IE paradigms has their own strengths and weaknesses, therefore, raising the question of what is the trade-off between NLP depth (and associated computational cost) versus effectiveness. We have used these two kinds of linguistic analysis in our method. Actually, the constituent extractors in the hybrid methods are based on shallow and deep linguistic analysis tools. The shallow extractors which used are TextRunner [40] (based on a lightweight noun phrase chunker and part-of-speech tagger) and ReVerb [16] (mostly based on matching patterns over part-of-speech tags). They are integrated with DepOE [22] (based on dependency parser). More details about these Open IE systems were presented in Sect. 2. TextRunner [40] is the first approach for Open IE, which has achieved relative high performance in handling an unbounded number of relations. TextRunner's shallow linguistic features lead to scalable but noisy extractions. Therefore, its performance measures are low. Dependency parsing gives unambiguous syntactic relationships among each word in the sentence and identifies subtrees connecting the relation trigger and its arguments. It could improve precision and recall over shallow syntactic features, but at the cost of extraction speed [22].

One of the major criticisms within the Open IE paradigm against deep extractors (such as dependency-based methods) concerns the computational cost associated with rich syntactic features. In our opinion, a proper technique for open relation extraction should be rapid enough in order to handle a very large amount of data efficiently while keeping performance measures in appropriate level. According to this view, high performance subset of shallow Open IE methods' outputs was examined.

According to [15] TextRunner can extract relations with high performance when sentence length is short. Its performance deteriorates rapidly as sentences get longer. TextRunner's precision is high when the sentence length is short. This is because long sentences tend to have long-distance relations which are difficult to be handled with shallow features. Considering the features selected by TextRunner, it does not attempt to model long-distance relationships [14, 15]. This correlation between performance of TextRunner and sentence length is useful and provides insight into the strength of TextRunner's shallow syntactic processing, and leads us to construct a high precision subset of TextRunner, which can be employed in our proposed approach. On the other hand, deep extractors can handle long-distance relations in long sentences and thus are proper for long sentences. These properties were used in the proposed sentence-based hybrid approach. Combination of these subsets of shallow and deep systems leads to have a high performance sentence-based hybrid method. Due to the use of shallow extractors, the proposed approach will take less time in comparison with the pure deep extractors.

In fact, the above mentioned advantages of shallow extractors motivate us to focus on improving defects of these extractors. For this reason, we try to keep the strengths of them and improve them by using deep extractors. Parser features help to handle complicated and long distance relations in difficult sentences [15]. For this purpose, we exploit a parser-based deep Open IE extractor namely DepOE [22] and try to combine it with TextRunner. With respect to these cases, a hybrid Open IE paradigm by incorporating strengths of TextRunner and DepOE was suggested. The outline of proposed hybrid method based on sentence length is shown in Fig. 1.

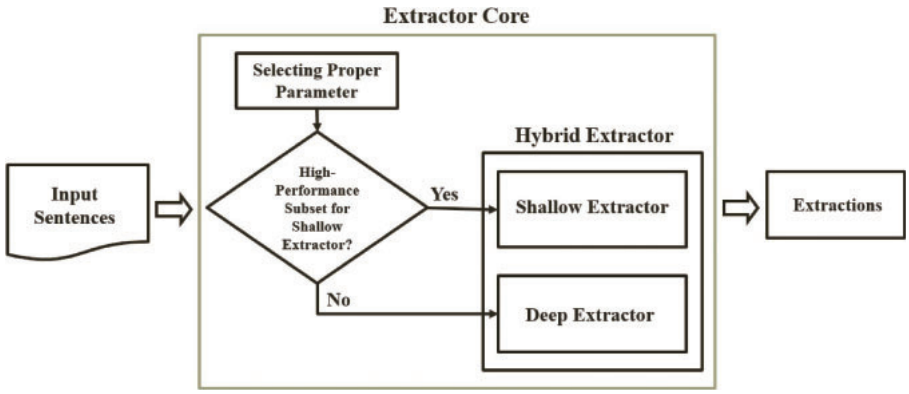


Fig. 1. The outline of proposed hybrid method based on sentence length.

TextRunner with DepOE system is combined and a hybrid system is generated to cover the aforementioned limitation of TextRunner. For this purpose, two main parameters are used: sentence length and confidence measure. Different values of sentence length are exploited. TextRunner is applied to the sentences with lengths equal or less than specified length threshold and DepOE is applied to the sentences with the length of more than specified length threshold. Final extractions are determined by taking union of these two extractor outputs. This method is called TextRunner-DepOE$_{SL}$ (TR-DOE$_{SL}$ which SL subscript is used for Sentence Length).

Since the probability of a low confidence extraction being a correct instance of relation among arguments is low, a configuration of proposed hybrid approach is also run, in which variations of shallow extractors' confidence parameter are considered and deep extractor is applied to the sentences of extractions with confidence values equal to or lower than a specified threshold. Final extractions are determined by taking union of these two extractors' outputs. We examine performance of the hybrid approach over different threshold values and detect the best trade-off between precision and recall by tuning confidence measure as a combination parameter. Confidence-based hybrid systems increase precision by filtering out extractions that are expected to have low precision. Shallow extractors' precision is low, when the confidence is low. Actually, these methods focus on increasing the precision and use the confidence as a filter policy to decrease the number of incorrect extractions and increase the correct ones, as a result, lead to high performance. We first keep TextRunner extractions with confidence values equal to or above a threshold and then run DepOE over all sentences yielding lower confidence than that threshold and take the union of these two extractions. We refer to this version as TextRunner-DepOE$_{C}$ (TR-DOE$_{C}$ which C subscript is used for Confidence measure).

In this case, the proposed hybrid method changes slightly. The main structure of the proposed confidence-based hybrid method has been shown in Fig. 2. These methods were run for a range of threshold values and the results were reported in Sect. 4.

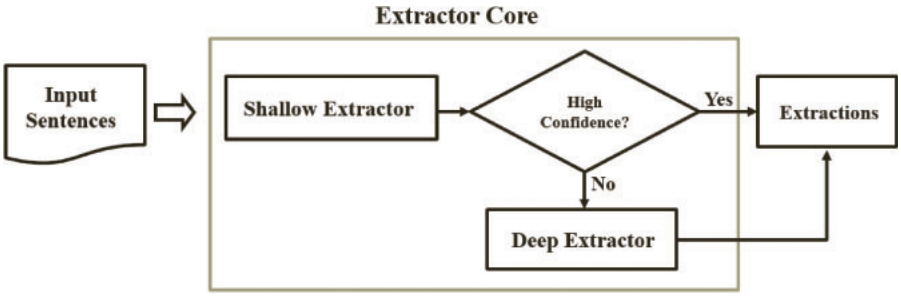


Fig. 2. The outline of proposed hybrid method based on confidence measure.

Another hybrid system is also presented that combines ReVerb with DepOE. ReVerb [17] is considered to be one of the best robust shallow Open IE extractors up to now. This version is referred to as ReVerb-DepOE (RV-DOE). ReVerb in contrast to TextRunner, does not require any training data. Thus, RV-DOE's advantage over TR-DOE is that it is completely independent of any training data. Regarding F-measure, DepOE performs better than ReVerb in the full extraction task. It is expected that incorporating ReVerb with DepOE would achieve higher performance than both systems individually.

Unlike TextRunner, the efficiency of ReVerb on short sentences is not clear and we will present a full analysis thereof in Sect. 4. Different threshold values of confidence and sentence length are applied (RV-DOE$_{SL}$ and RV-DOE$_{C}$) and the impact of these factors on resulting performance is assessed.

4. Experimental Results

We evaluate the effect of applying DepOE, a deep extractor, on the weaknesses of TextRunner and ReVerb with the aid of two parameters (namely, sentence length and confidence measure) and explore the behavior of these shallow extractors. We used the dataset that was provided by Fader and his colleagues [16] in our experiments. This dataset has some advantages because of its large sample size (more than twice of several other datasets such as [22]). Another advantage is that it is a standard dataset which has been used in several important recent studies such as [13, 20, 25, 33, 41]. They created a test set of 500 sentences sampled from the Web, using Yahoo's random link service. This dataset contains the output of the different extractors run (such as TextRunner and ReVerb) on the 500 selected sentences. Two human judges independently evaluated each extraction as 'correct' or 'incorrect'. The judges reached agreement on 86% of the extractions, with an agreement score of $\kappa = 0.68$. The subset of the data where the two judges concur is used in our experiments. The judges labeled uninformative extractions (where critical information was dropped from the extraction) as incorrect. This is a stricter standard than was used in previous Open IE evaluations [17].

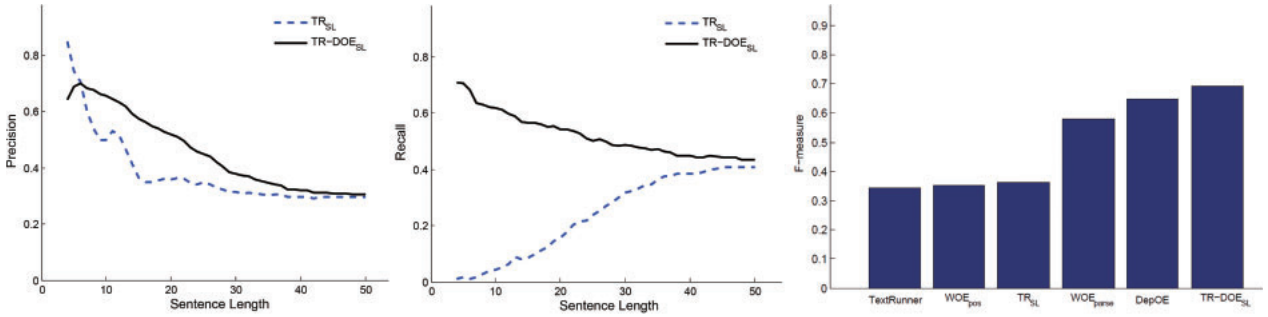
Since recent systems such as DepOE [22] were not exist in the dataset, there is a need for making a slight modification in dataset. We do the same evaluation steps for DepOE extractions. DepOE is run over the input sentences. Each resulting relation was independently examined and labeled as 'correct' or 'incorrect' by two people. The annotators reached agreement on about 95% of DepOE extractions, with an agreement score of $\kappa = 0.75$. We use the subset of the data where the two human annotators reached an agreement on the labels. The subset of the sentences which the judges reached agreement on the extractions taken from them was used in the experiments.

As mentioned before, a hybrid system is presented that incorporates TextRunner with DepOE system in order to cover the aforementioned limitation of TextRunner as a shallow Open IE system. Different values of sentence length are exploited and TR-DOE$_{SL}$ is described in detail in this section. TextRunner was also run over sentences equal or shorter than specified length threshold and remained sentences were discarded. This version is called TextRunner$_{SL}$ (TR$_{SL}$). These methods are run for a range of threshold values and the results are reported in Fig. 3. The performance variations of these methods are evaluated and compared. For a given threshold, the precision and recall can be measured. Precision is the ratio of the number of correct extractions retrieved to the total number of extractions retrieved. Recall is the ratio of the number of correct extractions retrieved to the total number of correct extractions in the dataset. Total number of extractions from two systems labeled as correct by the judges were used as our measure of recall for the corpus, as done in prior Open IE systems such as ReVerb [16].

Figure 3 reports the results from an analysis for all sentence lengths in dataset. TR$_{SL}$ and TR-DOE$_{SL}$'s precision curves are compared in Fig. 3(a). TR$_{SL}$ and TR-DOE$_{SL}$ both have good precision on short sentences. Except for short sentences (those with lengths between 4 and 8 words), the precision of TR-DOE$_{SL}$ is always higher than that of TR$_{SL}$. TextRunner's precision deteriorates faster than TR-DOE$_{SL}$ as sentences get longer. This can be caused by locality phenomenon. Locality is defined as the number of tokens between the arguments in a sentence. According to [23] TextRunner's shallow features can extract relations more reliably when locality is low. TextRunner precision is high when the locality is low. In low locality, TextRunner even is able to obtain higher precision than semantic role labeling-based extractors [23] which perform deep syntactic and semantic analysis. Here it seems that, locality phenomenon is reinforced in short sentences. This correlation between precision of TextRunner and locality is useful and provides insight into the strength of TextRunner's shallow syntactic processing, and leads us to construct a high precision subset of TextRunner, which can be employed in our proposed hybrid approach. TR-DOE$_{SL}$ starts at high precision due to using a high precision subset of TextRunner (short sentences) and discarding of potentially low quality extractions from it. Some part of dataset with longer sentences which is used by DepOE has high precision too. This is mainly because parser features used by DepOE are useful for handling long sentences and thus, overall precision of TR-DOE is high. Finally, about all of dataset's sentences are used by TextRunner in high sentence length thresholds and the TR-DOE$_{SL}$'s curve tends to approach the precision of TextRunner. Locality increases as sentence length increases, thus the precision of the TR$_{SL}$'s curve decreases as sentence length increases and finally, the two curves reach the same asymptotic level.

Figure 3(b) shows that TR-DOE$_{SL}$'s recall ranges from about the recall of DepOE to about the recall of TextRunner. It is high initially, because large amount of dataset is exploited by DepOE and thus recall of TR-DOE$_{SL}$ is near the recall of DepOE. Since the number of sentences used by TextRunner is few initially, the TR-DOE$_{SL}$'s recall is directly affected by DepOE's recall. It decreases gradually as sentence length increases. Since about all of sentences are exploited by TextRunner in high sentence length thresholds, total recall reaches near TextRunner's level. TR$_{SL}$'s recall variations with sentence length are also shown in Fig. 3(b). TR-DOE$_{SL}$'s recall curve has an opposite direction when compared to TR$_{SL}$. This case reflects the nature of recall. TR$_{SL}$'s recall is low when the number of sentences is low. When the sentence length threshold is high, so is the number of sentences used by TR$_{SL}$ and thus recall is high.

In sentence-based hybrid methods, the value of the threshold was examined from 3 to 50 by step 1. We examined F-measure values for all thresholds and found the maximum amount of it for each of the two methods. The F-measure is the uniformly weighted harmonic mean of the precision and the recall. Determination of the maximum value for F-measure is an attempt to find a best possible trade-off between recall and precision. It was observed that the extractions with the sentence length thresholds of lower than 8 words and the sentence length thresholds of higher than 43 words yield maximum F-measure for TR-DOE$_{SL}$ and TR$_{SL}$ respectively and TR-DOE$_{SL}$ has almost two times more F-measure than TextRunner. TR-DOE$_{SL}$ also achieves best F-measure in comparison with the other strategies such as WOE$_{pos}$ and WOE$_{parse}$. This result shows that TR-DOE$_{SL}$'s performance in short sentences is higher than that of DepOE itself.

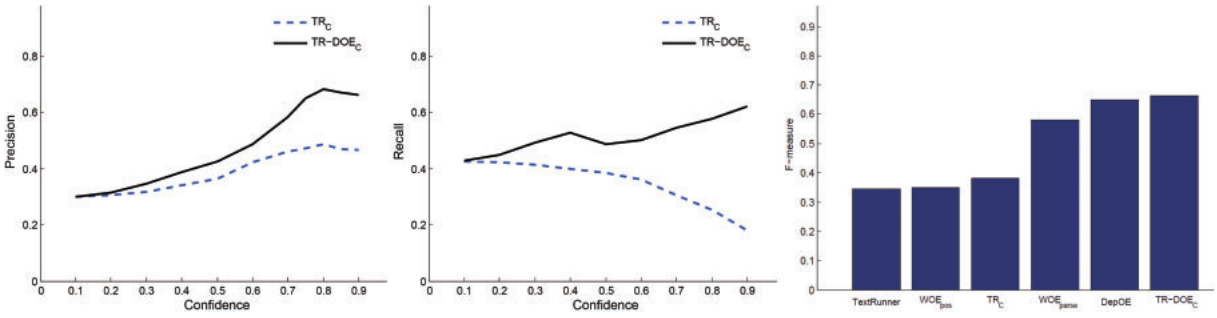


(a) (b) (c)

Fig. 3. (a) TR$_{SL}$'s precision starts higher than the precision of TR-DOE$_{SL}$. TR-DOE$_{SL}$ has high precision at low sentence length (b) Recall rises over sentence length for TR$_{SL}$. TR-DOE$_{SL}$ has high recall at low sentence length (c) TR-DOE$_{SL}$ has an F-measure almost twice that of TR$_S$ and TextRunner.

A configuration of the proposed hybrid approach is also run, in which variations of shallow extractors' confidence parameter are considered (TR-DOE$_C$). Another method which is called TextRunner$_C$ (TR$_C$) focuses on using only extractions with confidence values equal or above a threshold and ignores other extractions. Figure 4(a) shows variations of precision for different values of TextRunner's confidence measure. TR-DOE$_C$'s precision is near the precision of TextRunner when the confidence threshold is low and the number of sentences used by DepOE is few. As confidence grows, the DepOE is applied to more sentences and the precision becomes higher. When confidence increases, TR$_C$'s precision also increases as a result of filtering incorrect extractions. Figure 4(a) shows that the slope of TR-DOE$_C$'s precision curve increases quicker than that of TR$_C$, due to deep features used in DepOE. TR-DOE$_C$ extracts accurate triples and the precision of TR-DOE$_C$ is better than TR$_C$. The recall curves are shown in Fig. 4(b). While TR-DOE$_C$'s recall improves on TR$_C$'s recall, TR$_C$ loses recall slowly. TR$_C$'s recall curve drops rapidly, as more confidence is included. Since many relations have confidences above the low confidence thresholds, DepOE is used for less sentences, hence the resulting recall of TR-DOE$_C$ is low initially, and it is near the recall of TextRunner. As confidence threshold values increase, the recall gradually increases and reaches nearly to the level of DepOE's recall. This is because of the major impact of DepOE system. TR$_C$ starts at slightly higher recall due to considering about all of TextRunner true extractions. Unlike TR-DOE$_C$, TR$_C$'s recall decreases as the values of confidence threshold increase. When confidence is high, the number of missed correct extractions increases and hence the recall becomes low. Figure 4(b) shows that the recall of TR-DOE$_C$ is significantly higher than TR$_C$.

In confidence-based hybrid methods, the value of the threshold was examined from 0.1 to 0.9 by increments of 0.1. We evaluated F-measure for TextRunner and TR-DOE$_C$ in the same way as we did before. The highest F-measure for each of these two systems in comparison with that in the other systems is shown in Fig. 4(c). It was observed that the extractions with the confidence threshold of higher than 0.8 and extractions with the confidence threshold of 0.6 achieve the highest F-measure for TR-DOE$_C$ and TR$_C$ respectively. The proposed hybrid method provides significant boost in performance of the shallow extractors. TR-DOE$_C$ achieves an F-measure that is slightly higher than DepOE and has double the F-measure of TextRunner or TR$_C$. TextRunner and TR$_C$ produce relatively identical F-measures. As shown in Fig. 4(c), the F-measure obtained by TR-DOE$_C$ was higher than that of any other method. This result shows that TR-DOE$_C$'s performance in high confidence values is higher than performance of DepOE itself.

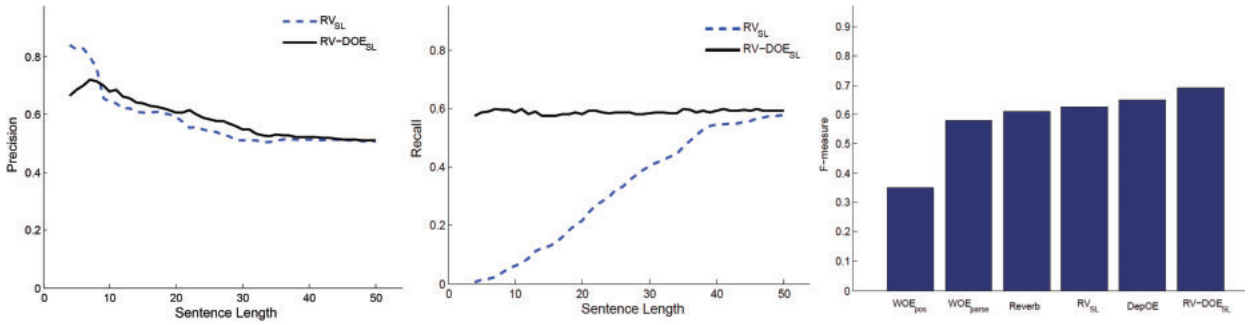


(a) (b) (c)

Fig. 4. (a) Precision rises over confidence for both systems, with TR-DOE$_C$ ending higher. (b) TR-DOE$_C$ reaches high recall over high confidence. (c) TR-DOE$_C$ system outperforms others.

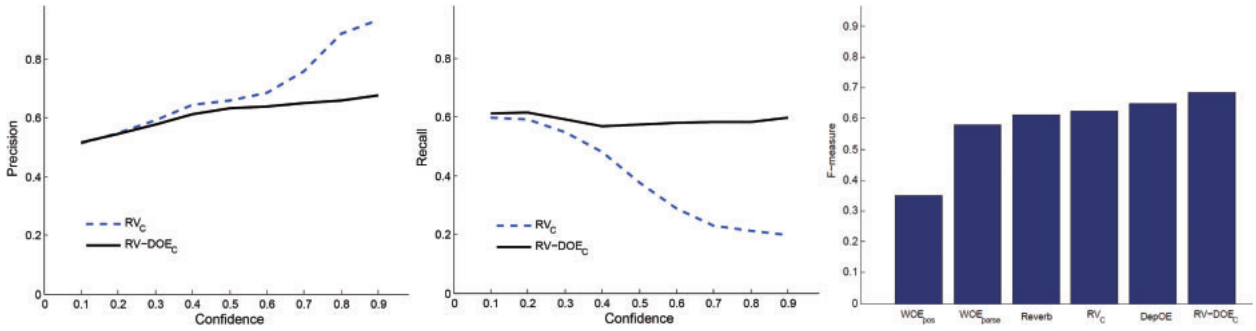
We now present RV-DOE hybrid system that combines ReVerb with DepOE. Like Textrunner, ReVerb was also run over sentences equal or shorter than specified length threshold (RV$_{SL}$) and remained sentences were discarded. Our experiments were set up identical to the ones we described in previous section. The results were summarized in Figs. 5 and 6. The results are similar to the results of previous investigations, but there are some differences. The precision and recall for RV$_{SL}$ and RV-DOE$_{SL}$ are shown in Fig. 5, again measured against sentence length. RV$_{SL}$'s precision curve shows that ReVerb's precision is high when sentence length threshold is low. The slope of precision curves decline at much slower rate than those for TR$_{SL}$ and TR-DOE$_{SL}$ and end at higher level than those for TR$_{SL}$ and TR-DOE$_{SL}$. Except for short sentences, the precision curves of RV-DOE$_{SL}$ and RV$_{SL}$ are close to each other. These are mainly because ReVerb focuses on identifying a more meaningful and informative relation phrase and improves both precision and recall in comparison to previous extractors such as TextRunner. Since DepOE has slightly lower recall than ReVerb, it is expected for RV-DOE$_{SL}$'s recall curve to have a quite low ascending slope. Figure 4(b) shows RV-DOE$_{SL}$'s curve obtains trivial recall over shorter sentences and then comes close to ReVerb's recall and stabilizes. This is mainly due to extracting relation in low sentence length thresholds by RV-DOE$_{SL}$ is affected by ReVerb's recall over short sentences. This case compensates trivial difference in the recall values of both systems. Since DepOE's recall is higher than TextRunner's recall, TR-DOE$_{SL}$'s recall is quite affected by DepOE's recall and the mentioned case does not occur [see Fig. 3(b)].

RV$_C$ focuses on using only extractions with confidence values equal or above a threshold and ignores other extractions. Figures 6(a) and 6(b) report the performance of RV-DOE$_C$ and RV$_C$ with different confidence thresholds. As these figures show, the results are similar to the outputs of TR-DOE$_C$ and TR$_C$. There are also some differences. RV-DOE$_C$ and RV$_C$'s precision curves have higher levels of precision than those of TR-DOE$_C$ and TR$_C$. This shows the effectiveness of ReVerb and its confidence measure. F-measure of four systems were compared and shown in Figs. 5(c) and 6(c). It was also observed that sentences with lengths shorter than about 9 words and sentences with lengths longer than 47 words yield maximum F-measure for RV-DOE$_{SL}$ and RV$_{SL}$ respectively and also extractions with confidence above 0.9 and extractions with confidence 0.4 achieve the highest F-measure for RV-DOE$_C$ and RV$_C$ respectively. Regarding F-measure, RV-DOE$_C$ performs better than ReVerb in high confidence values.



(a) (b) (c)

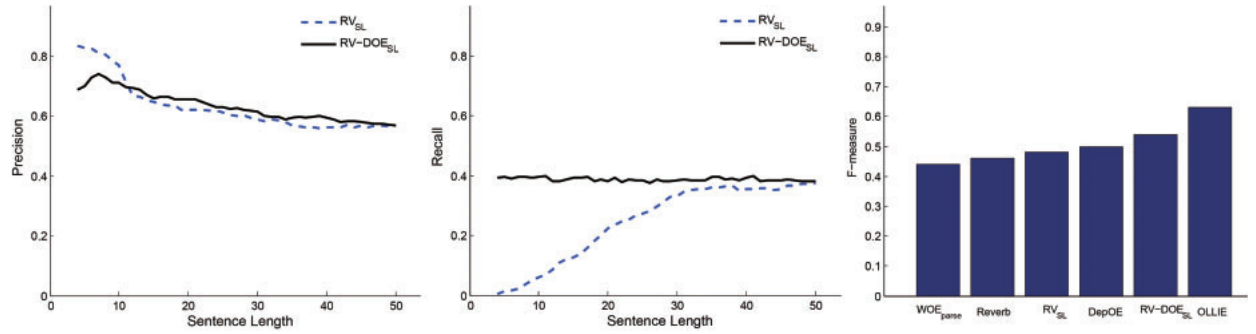
Fig. 5. (a) RV$_{SL}$'s precision starts higher than the precision of RV-DOE$_{SL}$. RV-DOE$_{SL}$ has high precision at low sentence length. (b) RV-DOE$_{SL}$ comes close to ReVerb's recall and is relatively steady. Recall rises over sentence length for RV$_{SL}$. (c) RV-DOE$_{SL}$ achieves better F-measure than others.



(a) (b) (c)

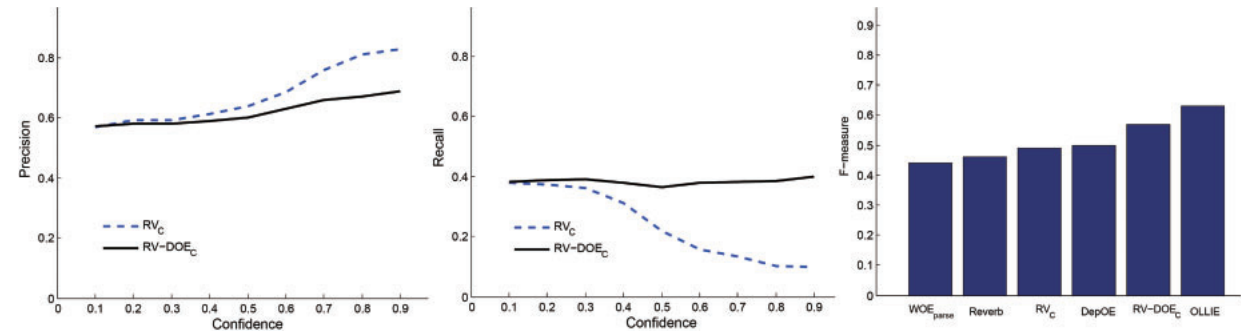
Fig. 6. (a) Precision rises over confidence for both systems, with RV_C ending higher. (b) RV-DOE's curve varies about between the recall of ReVerb and DepOE. (c) TR-DOE_C's F-measure is higher than ReVerb's.

We used the OLLIE dataset which is consists of 300 sentences with manually labeled extractions for ReVerb, WOE_{parse} and OLLIE. We compare RV-DOE to these systems and the results were reported in Figs. 7 and 8. The orientation of figures is the same as Figs. 5 and 6 and thereby more details of them weren't investigated here. OLLIE's ability to handle defects of ReVerb and DepOE made its performance better than others.



(a) (b) (c)

Fig. 7. (a) $RV\text{-}DOE_{SL}$ has high precision at low sentence length. RV_{SL}'s precision starts higher than the precision of $RV\text{-}DOE_{SL}$. (b) $RV\text{-}DOE_{SL}$ comes close to ReVerb's recall and is relatively steady. Recall grows over sentence length for RV_{SL}. (c) $RV\text{-}DOE_{SL}$ achieves better F-measure than its constituent systems.



(a) (b) (c)

Fig. 8. (a) Precision rises over confidence for both systems, with RV_C ending higher. (b) RV-DOE's curve varies about between the recall of ReVerb and DepOE. (c) TR-DOE_C's F-measure is higher than ReVerb's.

4.1 Error analysis of proposed method

There are some cases where hybrid approach can improve results due to integration of its constituent systems. Some facts can be correctly obtained from both constituent extractors of the hybrid method. In this case, combination parameter leads the hybrid method to use shallow extractor instead of deep one. Therefore, computational time overhead is reduced and speed is increased. For example all of the extractors can extract (*His teammates, loved, the*

show) from "*His teammates loved the show*". Using the sentence length parameter causes ReVerb to process this sentence and obvious improvement on the performance of the approach is achieved. This demonstrates the usefulness of the proposed combination parameter.

In most cases, the hybrid system distributes input sentences to proper extractors and therefore the desired result is achieved. The most common error in the ReVerb Open IE system is incorrect identification of the first argument (arg1). DepOE has more precise extractions of the two arguments, in particular of arg1, since the parser is able to correctly identify the subject. For example ReVerb extracts (*Dartmoor National Park, was followed by, hikers*) from the sentence "*the path of river through Dartmoor National Park was followed by hikers.*" while DepOE extracts (*The path of river through Dartmoor National Park, was followed by, hikers*). A common error of ReVerb is extraction of only a truncated part of the second argument (arg2). For example, in the sentence "*The method would increase the count of non-SPAM messages which are classified as spam for the given domain.*" arg2 gets truncated and ReVerb extracts (*The method, would increase the count of, non-SPAM messages*). In contrast to ReVerb, DepOE performs well on the second argument and extracts (*The method, would increase, the count of non-SPAM messages which are classified as spam for the given domain*). In these examples, according to the combination parameter, DepOE is selected by the hybrid method and incorrect extractions of ReVerb are prevented from occurring, but this is not always the case. Despite satisfying the combination parameter condition, the proposed method sometimes cannot detect the arguments of relation correctly. We describe it in details in this section. Extracting correct arguments still remained an important problem, which is mainly related to weakness of the constituent extractors of the hybrid approach. We performed an analysis of our hybrid approach errors. We randomly sampled the 100 incorrect tuples from the dataset of Fader and his colleagues [16] and analyzed the errors. Table 1 summarizes the types of errors and the fractions of each type for RV-DOE$_{SL}$.

Sometimes, the weaknesses of these hybrid approaches are related to the extractors used in them and improvement of these methods depends on improving its constituent systems. In this case, the result is not affected by the combination parameter and even if the proper extractor is selected, correct facts cannot be detected. For example none of the constituent systems of the hybrid method can extract n-ary relations. We found that 27% of the incorrect extractions were to mistake an n-ary relation as a binary relation. For example, all three systems extract (*This program, gave, Iran*) from the sentence "*This program gave Iran its first group of professional nuclear engineers*". One possible solution is to deploy a deep extractor that has high capability to detect n-ary relations such as ClauseIE and KrakeN (see Sect. 2 for more detail). Basic components of hybrid approach can be selected based on reducing defects of information extraction systems so that the resulting system has high efficiency. In addition, it would be enriched by some beneficial properties of constituent extractor, for example in [37] an approach based on structured discourse graphs was proposed which can extract n-ary relations and support some relevant information in natural language texts which are irregular, highly contextualized, poorly structured, and intrinsically ambiguous. These characteristics are not usually supported by information extraction approaches. Semantic networks, can also be used as a research platform for elaborating Open IE methods and the construction of a hybrid method for extracting relations. They are n-ary and include potential context information, for example in [38] an approach for building large-scale semantic networks from text, employing deep semantic processing was presented (see Sect. 2 for more detail). IE approaches can be complemented by these scenarios and this is left as future work.

Another example is related to semantic meaning. Under some specific conditions (conditional subordinate clause or adverb), the semantic of the original sentence is affected. In this case, some extractions do not reflect the meaning conveyed by the sentence. For example, the systems extract (*Alex, only reads, book*) from the sentence "*Alex only reads book when he is in the train*".

Other sources of failure were due to non-contiguous relation phrases, imperative verb and other errors including conditional subordinate clauses, part-of-speech errors, overspecified relation phrase, etc. As mentioned, although the combination parameter works well mostly, there are some cases where the selected extractor does not work properly and the hybrid method should use the other constituent extractor instead. This problem is mainly related to the weakness of the shallow extractor and may be avoided by selecting appropriate constituent extractor of the hybrid method based on other effective factors. As our analysis shows 4% of the incorrect extractions are raised when relation components are far apart of each other. This case can happen when there is a phrasal verb in the sentence. For instance, in the sentence "*John turned the lights off*", Reverb extracts (*John, turned, the* lights) but DepOE extracts (*John, turned off, the lights*). In this example, the condition of the sentence length parameter for shallow extractor is satisfied but ReVerb does not yield correct triple in this case and using DepOE is more beneficial. Another kind of errors that can occur are imperative verb errors. They cause erroneous results. For example, Reverb extracts (*you, close, the door*) from the sentence "*Could you close the door?*". There are phrases that satisfy the syntactic constraint of ReVerb, but are not relational. ReVerb can sometimes match relation phrases that are so specific that they have only a few possible instances, even in a Web-scale corpus. This phenomenon usually happens in the condition of the longest match for a verb phrase. Consider the sentence: "*The Obama administration is offering only modest greenhouse gas reduction targets at the conference*". The relation will be the phrase: is offering only modest greenhouse gas reduction targets at. This phrase is specific for the argument pair (*Obama administration, conference*) [16]. This kind of errors shows that an error in relation detection leads to an error in argument detection.

Part-of-speech error is unavoidable, but one of the least common sources of errors. It occurs at the preprocessing stage and results in errors in extractions at a later stage. Even the state-of-the-art morphological analyzers commit errors in part-of-speech tagging [42]. For example, the word 'write' when used as a noun causes trouble. In the sentence "*This file has write permission*" part-of-speech ambiguity affects the result and ReVerb extracts (*This file, has write, permission*). Open IE systems have difficulty in cases where the part-of-speech of a word is ambiguous or difficult to tag automatically. RV-DOE$_C$'s incorrect extractions consist of incorrect extractions of ReVerb and DepOE, because they are generated from outputs of both systems.

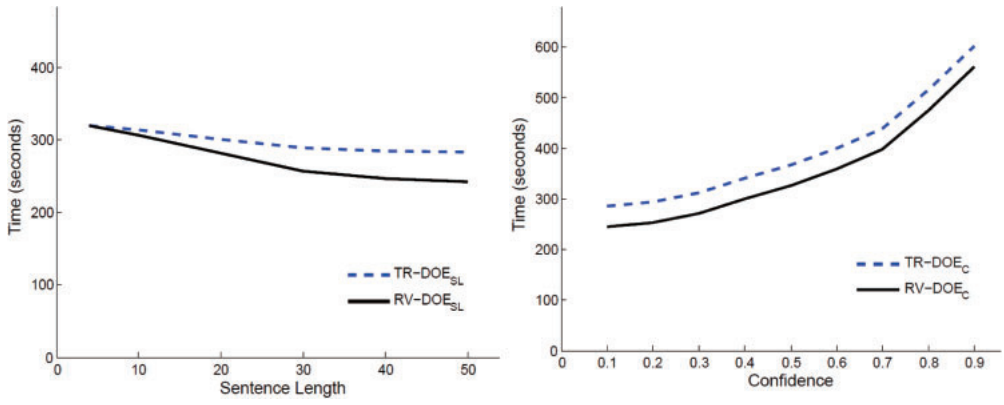
Table 1. Distribution of the types of errors in RV-DOE$_{SL}$.

62%	Incorrect arguments
27%	N-ary relation
4%	Non-contiguous relation phrases
2%	Imperative verb
5%	Other errors including conditional subordinate clauses, POS errors, overspecified relation phrase.

4.2 Extraction speed

The extraction speed of proposed hybrid systems was tested. To account for computational time, the amount of time taken to use each system should be estimated. We ran each extractor on a set of 100,000 sentences and used Java for implementing the extractors. We tested on a Linux platform with a 2.4 GHz CPU and 2 GB memory. The processing time were 4 min and 2 s for ReVerb, 4 min and 43 s for TextRunner and 5 min and 19 s for DepOE. Since one of our goals was to have a high-speed system, we used a deep but fast Open IE system. DepOE has reasonable processing time and is based on robust and fast rule-based dependency parsing. There is always a trade-off between performance and speed when selecting a deep extractor. Deep extractors usually have high computation cost. In general, the deeper the extractor, the higher is the incurred computation cost.

Figure 9(a) shows extraction time *vs.* sentence length. Hybrid extractors' processing time varies with different values of thresholds. If the number of short sentences is low then the hybrid extractors (TR-DOE$_{SL}$ and RV-DOE$_{SL}$) reduce to DepOE and the total time approaches the processing time of DepOE, resulting in a slight increase in total processing time. When the number of short sentences is high, hybrid extractors reduce to TextRunner/ReVerb and the total time approaches to the processing time of TextRunner/ReVerb, resulting in a slight decrease in total processing time. In worst case, hybrid extractors which are based on sentence length will approximately have the same processing time as DepOE. The beginning and the end of TR-DOE$_{SL}$ and RV-DOE$_{SL}$'s curves are corresponded to DepOE and TextRunner/ReVerb respectively. Similar cases also occur for the confidence threshold in TR-DOE$_C$ and RV-DOE$_C$. The difference is that the worst processing time for TR-DOE$_C$ and RV-DOE$_C$ is approximately attained by taking the union of each of their constituent extractors' processing time.



(a) (b)

Fig. 9. (a) RV-DOE$_{SL}$'s time decreases more slowly with sentence length than TR-DOE$_{SL}$ because ReVerb has lower processing time. (b) Time curves rise over confidence for TR-DOE$_C$ and RV-DOE$_C$.

4.3 Properties of hybrid extractors

The best performance for sentence-based hybrid methods was achieved using shallow extractors for short sentences and deep extractors for long sentences. As it was shown and discussed in detail, sentence-based hybrid methods cause increasing of accuracy and thus performance, in less (or at most equal) time than taken by its constituent deep extractor. These cases are important in large inputs such as Web-scale data. This method has successful results when the number of short sentences are high and we need a high performance extractor that utilizes the available time efficiently. By keeping extractions with high confidence and applying deep extractors for sentences yielded low confidence, confidence-based hybrid methods are able to achieve high performance. According to the detailed analysis, confidence-based hybrid methods can also improve precision and thus improve F-measure. Since they have time overhead, the employed deep method is fast enough in order to decrease total time as far as possible. The deeper the extractor, the higher is the performance. There is a trade-off between performance and time when selecting a deep extractor. These methods can be designed and implemented well by selecting appropriate deep extractor based on the available time.

Sentence-based and confidence-based hybrid methods are the same in their constituent systems. As bar charts show, their F-measure values do not have any significant difference. Their main difference is in their speed. As mentioned before, the aim of Open IE is to scale information extraction methods to the size and diversity of the Web domain. Although F-measures of both hybrid methods are higher than their constituent systems, sentence-based hybrid methods are much faster than confidence-based ones. This is an advantage of them over the confidence-based hybrid methods.

5. Conclusion

In this paper, we investigate the capability of using benefits of deep extractors for improving performance of shallow Open IE systems' extractions. We combined two shallow Open IE systems (namely TextRunner and ReVerb) individually with DepOE (a deep but fast processing Open IE system) and examined the performance of resulting hybrid systems. The variation of performance of proposed hybrid methods with sentence length and confidence parameters were also examined and the best trade-off between precision and recall was found. Experiment results demonstrate that our hybrid extractors achieve better F-measure in comparison with their constituent systems. TR-DOE has an F-measure almost twice than that of TR and TextRunner. In summary, we found that the best performance for TR-DOE_{SL} and RV-DOE_{SL} was achieved using shallow extractors for short sentences and deep extractors for long sentences. By keeping extractions with high confidence and applying deep extractors for sentences yielding low confidence, TR-DOE_{C} and RV-DOE_{C} are able to achieve high performance. Open IE needs speed over deeper processing, which aids in scaling to Web-scale corpora. The proposed hybrid approaches, in particular sentence-based hybrid methods, are rapid due to applying fast processing extractors and thus can reinforce scalability feature in pure deep Open IE systems. This research also shows that we can have a high performance hybrid method by applying deep extractor on the smart subset of the shallow extractor input.

The weaknesses of these hybrid approaches are related to the extractors used in them and improvement of these methods depends on improving its constituent systems. Several successful deep analysis based approaches which have higher performance than DepOE were proposed recently. It is expected that using systems stronger than DepOE will yield an improvement in performance. Although some deep extractors are unable to operate at the Web-scale individually, scalable hybrid systems may be able to use them in their structures.

REFERENCES

[1] Nakashole, "Automatic extraction of facts, relations, and entities for web-scale knowledge base population." (2013).
[2] Aggarwal and Zhai, "Mining text data." (2012).
[3] Min, Relation Extraction with Weak Supervision and Distributional Semantics. (2013).
[4] Yao, Haghighi, Riedel and McCallum, "Structured relation discovery using generative models", In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1456–1466 (2011).
[5] Piskorski and Yangarber, "Information extraction: Past, present and future". Multi-source, Multilingual Information Extraction and Summarization. Springer, pp. 23–49 (2013).
[6] Min, Shi, Grishman and Lin, "Towards Large-Scale Unsupervised Relation Extraction from the Web." *International Journal on Semantic Web and Information Systems (IJSWIS)*, **8**:(3): 1–23, (2012).
[7] Bunescu and Mooney, "A shortest path dependency kernel for relation extraction", In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 724–731 (2005).
[8] Culotta, McCallum and Betz, "Integrating probabilistic extraction models and data mining to discover relations and patterns in text", In Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, pp. 296–303 (2006).
[9] Kambhatla, "Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations", In Proceedings of the ACL 2004 on Interactive poster and demonstration sessions, pp. 22 (2004).
[10] Banko, Etzioni and Center, "The Tradeoffs Between Open and Traditional Relation Extraction", In ACL, pp. 28–36 (2008).
[11] Schmitz, Bart, Soderland and Etzioni, "Open language learning for information extraction", In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 523–

534 (2012).
[12] Soderland, Roof, Qin, Xu and Etzioni, "Adapting open information extraction to domain-specific relations." *AI Magazine*, **31**:(3): 93–102, (2010).
[13] Del Corro and Gemulla, "ClausIE: clause-based open information extraction", In Proceedings of the 22nd international conference on World Wide Web, pp. 355–366 (2013).
[14] Banko, Cafarella, Soderland, Broadhead and Etzioni, "Open information extraction for the web", In IJCAI, pp. 2670–2676 (2007).
[15] Wu and Weld, "Open information extraction using Wikipedia", In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pp. 118–127 (2010).
[16] Fader, Soderland and Etzioni, "Identifying relations for open information extraction", In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1535–1545 (2011).
[17] Etzioni, Fader, Christensen, Soderland and Mausam, "Open Information Extraction: The Second Generation", In IJCAI, pp. 3–10 (2011).
[18] Nebot and Berlanga, "Exploiting semantic annotations for open information extraction: an experience in the biomedical domain." *Knowledge and information Systems*, **38**:(2): 365–389, (2014).
[19] Akbik and Broß, "Wanderlust: Extracting semantic relations from natural language text using dependency grammar patterns", In WWW Workshop (2009).
[20] Akbik and Löser, "Kraken: N-ary facts in open information extraction", In Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction, pp. 52–56 (2012).
[21] Qiu and Zhang, "Zore: A syntax-based system for chinese open relation extraction", In Proceedings of EMNLP (2014).
[22] Gamallo, Garcia and Fernández-Lanza, "Dependency-based open information extraction", In Proceedings of the Joint Workshop on Unsupervised and Semi-Supervised Learning in NLP, pp. 10–18 (2012).
[23] Christensen, Soderland and Etzioni, "An analysis of open information extraction based on semantic role labeling", In Proceedings of the sixth international conference on Knowledge capture, pp. 113–120 (2011).
[24] Kim and Compton, "Improving open information extraction for informal web documents with ripple-down rules". Knowledge Management and Acquisition for Intelligent Systems. Springer, pp. 160–174 (2012).
[25] Tseng, Lee, Lin, Liao, Liu, Chen, Etzioni and Fader, "Chinese open relation extraction for knowledge acquisition." *EACL 2014*: 12, (2014).
[26] Crysmann, Frank, Kiefer, Müller, Neumann, Piskorski, Schäfer, Siegel, Uszkoreit and Xu, "An integrated architecture for shallow and deep processing", In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp. 441–448 (2002).
[27] Mesquita, "Clustering techniques for open relation extraction", In Proceedings of the on SIGMOD/PODS 2012 PhD Symposium, pp. 27–32 (2012).
[28] Castella Xavier, de Lima, Lúcia and Souza, "Open information extraction based on lexical-syntactic patterns", In Intelligent Systems (BRACIS), 2013 Brazilian Conference on, pp. 189–194 (2013).
[29] Cimiano and Wenderoth, "Automatically learning qualia structures from the web", In Proceedings of the ACL-SIGLEX workshop on deep lexical acquisition, pp. 28–37 (2005).
[30] Mesquita, Schmidek and Barbosa, "Effectiveness and efficiency of open relation extraction." *New York Times*, **500**: 150, (2013).
[31] Bast and Haussmann, "Open information extraction via contextual sentence decomposition", In Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on, pp. 154–159 (2013).
[32] Bast and Haussmann, "More informative open information extraction via simple inference". Advances in information retrieval. Springer, pp. 585–590 (2014).
[33] Xu, Kim, Quinn, Goebel and Barbosa, "Open Information Extraction with Tree Kernels", In HLT-NAACL, pp. 868–877 (2013).
[34] Punyakanok, Roth and Yih, "The importance of syntactic parsing and inference in semantic role labeling." *Computational Linguistics*, **34**:(2): 257–287, (2008).
[35] Johansson and Nugues, "The effect of syntactic representation on semantic role labeling", In Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1, pp. 393–400 (2008).
[36] Franklin, Halevy and Maier, "From databases to dataspaces: a new abstraction for information management." *ACM Sigmod Record*, **34**:(4): 27–33, (2005).
[37] Freitas, Carvalho, Da Silva, O'Riain and Curry, "A Semantic Best-Effort Approach for Extracting Structured Discourse Graphs from Wikipedia", In Proc. of the 1st Workshop on the Web of Linked Entities, (ISWC) (2012).
[38] Wojtinnek, Harrington, Rudolph and Pulman, "Conceptual knowledge acquisition using automatically generated large-scale semantic networks". Conceptual Structures: From Information to Intelligence. Springer, pp. 203–206 (2010).
[39] Ebadat, Claveau and Sébillot, "Using shallow linguistic features for relation extraction in bio-medical texts." *Traitement Automatique des Langues Naturelles*: 125, (2011).
[40] Etzioni, Banko, Soderland and Weld, "Open information extraction from the web." *Communications of the ACM*, **51**:(12): 68–74, (2008).
[41] Xavier and Lima, "Boosting open information extraction with noun-based relations", In Proceedings of the ninth international conference on Language Resources and Evaluation (LREC'14). European Language Resources Association (ELRA), Reykjavik, Iceland, pp. 96–100 (2014).
[42] Zhila and Gelbukh, "Open Information Extraction from Real Internet Texts in Spanish Using Constraints over Part-Of-Speech Sequences: Problems of the Method, Their Causes, and Ways for Improvement." *Revista Signos. Estudios de Lingüística* **49**:(90), (March, 2016).