Big Data Knowledge Mining

Huda Umar Banuqitah , Fathy Eassa,Kamal Jambi, Maysoon Abulkhair
Computer Science
King AbdulAziz University
Jeddah, Saudi Arabia

***Abstract*—Big Data (BD) era has been arrived. The ascent of big data applications where information accumulation has grown beyond the ability of the present programming instrument to catch, manage and process within tolerable short time. The volume is not only the characteristic that defines big data, but also velocity, variety, and value. Many resources contain BD that should be processed. The biomedical research literature is one among many other domains that hides a rich knowledge. MEDLINE is a huge biomedical research database which remain a significantly underutilized source of biological information. Discovering the useful knowledge from such huge corpus leading to many problems related to the type of information such as the related concepts of the domain of texts and the semantic relationship associated with them. In this paper, an agent-based system of two–level for Self-supervised relation extraction from MEDLINE using Unified Medical Language System (UMLS) Knowledgebase, has been proposed . The model uses a Self-supervised Approach for Relation Extraction (RE) by constructing enhanced training examples using information from UMLS with hybrid text features. The model incorporates Apache Spark and HBase BD technologies with multiple data mining and machine learning technique with the Multi Agent System (MAS). The system shows a better result in comparison with the current state of the art and naïve approach in terms of Accuracy, Precision, Recall and F-score.**

Keywords—Knowledge Mining; Relation Extraction; Self-supervised; Big Data; Agent

I. INTRODUCTION AND BACKGROUND

Nowadays large spectrum data is being collected and generated on an unprecedented scale; this paradigm is called "Big Data"(BD)[1].In the last two decades, usage of biomedical computing systems present an explosive growth. The vast amount of Information they store, contains new knowledge that can provide decision support to improve the quality of medical care. MEDLINE is one example of the online bibliographic database on a biomedical domain that contains more than 22 million biomedicine journal articles[2]. As a result, these volumes of data require an efficient prediction and analysis platform to gain fast response and real-time classification for such BD[3]. The ability to discover knowledge from the big data in sufficient time and scalable fashion is a complex task. Data should be processed to extract some helpful knowledge from it. An essential challenge for applications of Big Data is that, the large volumes of data and extracts valuable information or knowledge for future actions[4]. The process of extracting useful knowledge from structured or unstructured data is known as knowledge discovery from Database (KDD) process which refers to a collection of activities designed to obtain new knowledge from complex data dataset[5][6]. KDD from such a biomedical corpus like MEDLINE is a complicated process, and it takes several processes [7]. Information Extraction (IE) techniques are the efficient exploitations of these resources that transform unstructured data into the structured form. An example of these techniques is Relation extraction (RE) which is an automatical mining of relations between the biomedical entities in text. The extraction of the relations between the biomedical entities is the procedure of determining the semantic link between those entities and characterizing the nature of this relationship [2]. Recently RE technigues has found growing interest amongst IE community and many studies concentrate on it because it helps to find new relations and interaction between biomedical entities from raw text and minimize usage of a human resource. RE includes multiple techniques such as Natural Language Processing (NLP), rule–based approach, and Machine Learning (ML) methods[8][9]. There are three types of RE approaches which are: Supervised that uses a corpus of labeled data, Unsupervised method which needs no labeling, and Self-supervised (distant-supervised) that uses a small set of labeled examples. The Unsupervised technique extracts strings of words that exist between the entities in huge amounts of text, and then simplifies and clusters these word strings to produce relation. Unsupervised methods can use massive quantities of data and extract very large numbers of relationships, but the resulting relations may not be simple to map to relations needed for a particular knowledge base.

Supervised relation extraction method, on the other hand, uses ML techniques to solve this problem. This approach requires a sufficiently annotated training data which consists of negative and positive examples. Moreover, the constructing of the annotated data set for training is expensive, time-consuming and requires expert knowledge. Self-supervised approach overcomes this problem by utilizing a knowledge base that includes informations about the exact target relation to automatic annotate the data set. The important assumptions are the sentences contain an entity pairs either represent or not represent a relation will also serve the relationship as well. On the other hand, Self-supervised approaches combine the advantages of supervised approaches, by including the features of noisy pattern in a probabilistic classifier, and Unsupervised methods, by extracting large numbers of relations from big corpora. It is generally believed that in a generic domain, Self- supervision techniques would benefit the relation extraction. However, in biomedical domain, the Self-supervised approach is not perfectly explored yet, becuase of two reasons. The first reason is that in general domain, the Freebase is the basic source of knowledge of Self

supervision technique, which is a lack of biomedical knowledge. The second is, the Self-supervison learning models assume that each entity instance is independent but in biomedical domain, this assumption is violated [10]. Thus a system model for Self-supervised Relation Extraction from Biomedical domain was proposed. As mentioned previously, KDD is iterative and interactive multiphase processes that include different steps like the selection of data , preparation and preprocessing, transformation of data , Data Mining (DM) and evaluation process. DM is the core process of KDD, and many researchers interested to integrate between DM and agents. DM can take benefit from agent through involving the intelligence to data mining system while the agents can take benefit from data mining through extending knowledge discovery capability of agents. There is some application that designs the process of KDD assimilates some modules to an agent, they proposed a strategy for integrating different techniques for mining database from agent perspectives [7]. For that, every module of the system was assigned to an agent to get the benefit of the agent technology in data mining process and improve overall system performance. diverse techniques for data mining have been integrated including Self-Supervision, natural language processing, machine learning and Multi-Agent System (MAS) to build a generalized Relation Extraction system from MEDLINE that requires minimal supervision using Unified Medical Language system (UMLS[1]).

The aim of this paper is to develop an agent based knowledge discovery system model for Self-supervised relation extraction in MEDLINE biomedical domain using UMLS knowledge base. Additionaly, different text features were implented with a paragraph to vector model and evaluate by using different classification algorithm to demonstrate the best algorithm with best features that can enhance the model performance for relation extraction. In addtion, Spark[2] and HBase[3] BD technology are integrated to speeding up the processing and accesing of such BD. The model has distinct two characteristics that distinguish the work from the existing ones which are: first, the construction of training example by using the semantic type of the concepts pair in MRREL section of UMLS is a new method of the exciting works in supervised relation extraction in the biomedical domain, and the second is using paragraph to vector model that transfer the sentence to vectors and using the resulted vectors as additional features with other features to improve the classifier performance; these characteristics improves the result comparing with others in terms of Accuracy, Precision, Recall and F-Score performance.

The rest of the paper organized as the follow. Section II presents the related work of the study, while section III describes the details of system architecture. Section IV introduces the methods used in the two levels of the proposed system and the experimental setup with the used dataset. Section V shows the results and the discussion while the final section is the conclusion.

II. Related Work

This section presents the different efforts that have been achieved in relation extraction from a biomedical domain which using distance supervised approach.

The author in [11] represents The general distant supervision approach for relationship extraction as following.

1) Identify a knowledge base which includes pairs of entities about the relationship-type in question (e.g., PPI-database).

2) Compile a large text (not annotated) resource relevant for the target domain (e.g., MEDLINE abstracts).

3) Recognize and normalize relevant named entities (e.g., protein names).

4) Associate entity-pairs from the knowledge base with previously identified instances in the text corpus.

5) Entity pairs contained in the knowledge base are labeled as positive instances. Negative instances are labeled by following the closed world assumption. The closed world assumption states that entity pairs lacking in the knowledge base do not feature the relationship type in question.

There are limited works which used Self-Supervised approaches in the biomedical domain. Most of these papers have used only the abstract of each paper, by utilizing the coordination structure of an entity in the sentences, [10] built up a Self-Supervised model which consolidates the result from open data extraction methodologies, to implement a task of relation extraction from biomedical resarch paper. They consider the structure coordination among entities that co-occurred in one sentence, is done by incorporate a grouping strategy to their model. They apply the Self-supervision technique to extract relationship of gene expression between genes and brain regions from literature. The Results showed that the model accomplish a better performance using Support Vector Machine (SVM) and with non-grouping strategy.

In [12] the authors trained the classifier using Self-supervision technique for Protein-Protein Interactions (PPI). They use a SVM classification algorithm as a classifier. IntAct database is the source of knowledge about interacting proteins.

Using UMLS as Knowledgebase, the authors in [13] proposed a Self-supervised approach for relation extraction from biomedical domain in MEDLINE abstracts using UMLS to annotate automatically the training data which is then used to train the classifier. To generate the training examples with positive and negative examples, all Concept Unique Identifier (CUI) pairs for the target relation are taken from MRREL and consider as a set of positive pairs. Hence, the presence of positive pair entities in a sentence will represent the target relationship. Any sets which additionally happen in another MRREL relations are expelled from the list of positive examples set. Conversely, negative instance will be detected depending on the positive pairs; new CUI pair combination will be created by joining all CUIs from the first position with all CUIs from the second position. These new combination will considered as a negative instance pair, only if a newly

[1] Unified Medical Language System (UMLS)
[2] http://spark.apache.org/
[3] http://hbase.apache.org/

produced CUI pair is not in the positive list and not appear in another MRREL relation. The model evaluated using two techniques Held-out and manual evaluation. On manual evaluation, the classifier was trained using the relation (may_treat), that created using Self-supervised and evaluated by using manually annotated corpus using test data set, and the result outperforms naïve approach with an F-Score of 0.571, 0.600 Precision and 0.545 Recall. The result indicated that UMLS is a useful resource for Self-supervised relation extraction. Additionally by utilizing UMLS to training a Self-supervised relation classifier,[14] exhibited the primary results utilizing UMLS knowledge base and the model assessed by utilizing the existing data set, since there were no directly annotated resources with UMLS relations is available. The presented model in [14] determined that utilizing a Self-supervised classifier which trained on MRREL relations like those found in the evaluation data set, will give propitious results.

The authors in [15] demonstrated the potential of Self-supervised learning in constructing a fully automated relation extraction process. They produced two distantly labeled corpora for drug to drug and protein–protein interaction extraction, with knowledge found in IntAct database for genes and Drug Bank database for drugs. They labeled approximately 50,000 MEDLINE abstracts using the shallow linguistic classifier trained on a distantly labeled corpus. In other words, the classifier trained on five manually annotated corpora and the same classifier trained on a distantly labeled corpus agree on 86.4 % of all 50,000 predictions.

There are some works done in Sel-Supervised approach outside the biomedical domain. Mintz and others in [16] provide relation extraction using Freebase for Self supervision. They utilized the same heuristic by matching tuples of Freebase with unstructured sentences from the Wikipedia articles in their experiments to produce features for learning relation extractors. instead of matching Wikipedia infobox with corresponding Wikipedia articles, matching Freebase with arbitrary sentences will potentially increase the size of matched sentences at the cost of accuracy. They conclude that their results suggest that syntactic features are quite useful in Self-supervised relation extraction. Also, the authors in [17] used Freebase knowledge base to annotate the corpus of New York Times with pairs of entity. They concentrated on the three basic relations which are birth place, nationality and contains. To prepare the classifier for training, they presented the utilization of a multi-instance learning technique for this context. In contrast, the authors in [18] annotated the information in the articles of Wikipedia using the infoboxes of Wikipedia as a knowledge source.

III. System Architecture

The system model consists of two main levels each with its own agents as shown in Fig1. The next subsections describe in details the components, functionality and the implementation of each level.

A. Level 1

The First level deals with data preparation and extraction, relation labeling with the usage of UMLS knowledge base, features extraction and training classifier on resulting train set.

1) Data preparation and extraction

MEDLINE corpus[4] is used as initial data. Medline is a large corpus of biomedical abstracts and articles. The sentences of MEDLINE contain the information of interest such as the biomedical entities. To use MEDLINE for the proposed Self-supervised system model, it should be annotated with these entities. And since UMLS KB was used to construct the training example in Self-supervised approach. So a mapping of UMLS concepts to the MEDLINE sentences is needed. For that, we used a MetaMapped MEDLINE, which is annotated by MetaMap tool[5]. Each sentence in MEDLINE annotated with UMLS concepts, and the annotations are represented in MetaMap machine output format[6].

UMLS is a collection of software and files that incorporate diverse biomedical knowledge base and vocabularies. Metathesasurus is a database in UMLS and contains a huge number of health and biomedical-related concepts and names and the relationship between them. all concepts arranged by their semantic type and all concept names are unified by Concept Unique Identifier (CUI). MRREL[7] form a small part of the Metathesasurus and includes diverse relations between various biomedical concepts which characterized by a couple of CUIs.

By following [13], tables from Metathesaurus have been used, which contains a mapping from Concept Unique Identifier (CUI) to Type Unique Identifier (TUI). MRREL defines binary relations between concepts, for that, a specific relations were used, these relations identified with "RO" keyword such as "may treat," "may prevent" and "gene product malfunction associated with disease". Those relations are most common for relation extraction task. Also, two semantic types have been used which are "**bacs**" and "**dsyn**" pairs, where "bacs" is Biologically Active Substance and "dsyn" is refer to Disease or Syndrome.

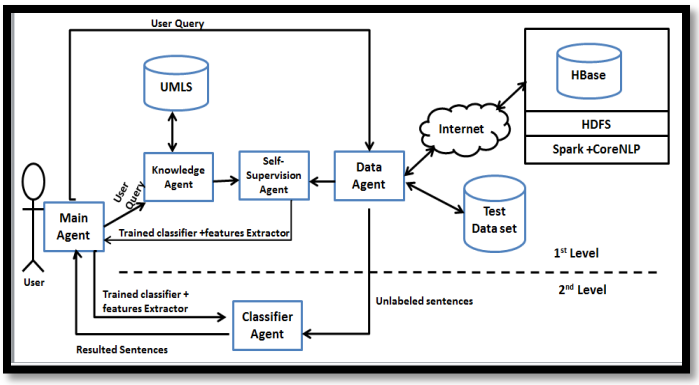

Fig. 1. System Architecture

[4] http://ii.nlm.nih.gov/MMBaseline/
[5] https://metamap.nlm.nih.gov
[6] https://metamap.nlm.nih.gov/Docs/2012_MMO.pdf
[7] MRREL table description

In the system framework, the main query to the knowledge base is taking all the relations between given semantic types. To make execution of the query fast, different tables from Metathesaurus have been joined to have all the needed information, so the resulting table contains pair (CUI1), (CUI2), relation and (TUI) of concepts.

The most time-consuming part in the system is getting sentences from Medline corpus that match the user query. Since the data size is too big to be handled by single commodity machine, advice from [6] was followed and stored these BD inside Hadoop distributed file system.

The main type of query in the system is getting all the sentences from Medline that has entities of a user-specified pair of UMLS semantic types. HBase is the only database that meets all the requirements. HBase is used to handle a large amount of data. It is designed to perform a fast linear scan on large collections, which can be used to perform fast queries.

To get the corpus in HBase, the files were located in HDFS file system. Then Spark workers have been run. Each worker takes a separate file and performs parsing, CoreNLP processing, and conversion to JSON format of each sentence. Hadoop and Spark help to do this job highly parallelizable – each small file can be processed independently.

2) Features Extraction

In the system model, the text are represented in multiple features to train the classifier by using two models, Bag of word model and paragraph to vector model. The description of each model with features will be in the following paragraphs:

a) Bag of world model features

Bag of world model is simple representation used in NLP. In this model the text such as sentences or document is represented as the bag of its words, disregarding of word semantic meaning or ordering in the text. The same text features were adopted, that depend on this model and implemented by [14] [16] [19] because they clearly represent the relation between the entities in the sentence also they help in determining the accurate class of the relation between disease and treatment. The adopted features are the sequence of words between entities, Post Of Speech tag (POS) of words between entities, Words on the semantic path between entities. For constructing these lexical and syntactic features,each sentence was annotated in the training set with part of speech tags and dependency tree using Stanford CoreNLP library[20], which has a large variety of instruments including parser, lemmatizer, tokenizer, part of speech tagger and it is written in Java programming language, which makes it easy to use inside Hadoop ecosystem.

a) Paragraph to vector model features

A paragraph to vector, or called (Doc2vec), is a method for constructing distributed vector representation for sentences and text documents [21]. In fact, the model "Doc2vec" has the potential to overcome many weaknesses of "bag-of-words" model. First, they inherit the most importan property of the word vectors: the semantics meaning of the words. The second advantage is that they take into consideration the word order considering word order and mapping semantically close words to close vectors. It was experimentally shown, that paragraph vectors can be better than other features for document classification task, this because the important characteristic of paragraph vectors is that they are learned from unlabeled data and thus can work well for the tasks that do not have enough labeled data. The paragraph or sentences in the model are mapped to a unique vector that can be used as features for the sentences; then these features can feed directly to conventional machine learning techniques such as logistic regression, support vector machines or others[21][22]. Therefore, from the previous characteristic of Doc2vec model, the model in [23] have been used to add the resulted vectors of sentences as addition features to represent the sentences and how the entities in the sentence related to each other, this to improve the relation extraction model.

3) Level 1 Agent Functionality

a) The main agent

interacts with system user and coordinates other agents

b) Knowledge agent

retrieves relations that correspond to a user query. Relations are represented as a triplet (CUI1, relation name, CUI2). Current knowledge agent implementation uses MRREL as a source of relations.

c) Data agent

finds sentence objects that correspond to a user query. Each sentence object contains the following information:

1) Id – unique identifier of a sentence, for Medline sentences it contains paper id.

2) Text – text of a sentence

3) Mappings – mappings from word to medical entity provided by Metamap tool. Mappings contain information about the semantic type, CUI, name and position in the text of matched entity.

4) Tokens – representation of CoreNLP parse results. Each token contains information about its POS, head, dependency relation, lemma and position in the text.

Data agent operation is chunk-based. When data agent receives a query, it gets result in a small chunk and transmits them to another agent one by one which helps to reduce total query time because other agents can start their work early.

a) Self-supervision agent

This agent constructs relation classifier with Self-supervision method. Self- supervision agent trains classifier. It constructs labeled training set without human intervention.

Self-supervision agent depends on knowledge and data agents. Knowledge agent provides relation examples, and data agent provides sentences and their parse results.

Self- supervision agent operates in several steps:

1) Constructs train set by matching sentence objects from data agent and relations from knowledge agent. This job is done chunk-wise. The result of this work is automatically labeled train set.

2) Fits feature extractors on train set and extracts features.

3) Trains classifier on extracted features.

4) Returns trained feature extractors and classifier to main agent.

Note that all relations are divided into two groups – 'general' and 'specific'. General relations most commonly are synonymous or 'is-a' relation. Specific relations represent more complex interactions between entities, for example, 'may treat' or 'may prevent'. Specific relations has label 'RO' in UMLS.

The following steps are the labeling train set algorithm:

1. For each sentence:
   a. Get all relations that have CUI1 and CUI2 same as CUI's of sentence entities
   b. If all matched relations are general – label sentence as negative example ("other")
   c. Else if sentence matched several specific relations or matched no relations – filter it out
   d. If sentence matched single specific relation and none of general relations – label it with specific relation
2. If some relations represent less than 5 % of train set – filter them out.

After labeling, Self- supervision agent performs feature extraction and classifier training. Then these extractors and classifier are sent to main agent.

B. Level 2

The second Level applies trained classifier to new data to label the unlabeled sentences.

1) Level 2 Agent Functionality

a) Classifier agent

This agent receives trained feature extractors and classifier from main agent. Then it gets an unlabeled sentence in chunks from data agent. For each chunk, it extracts features and performs label prediction with the classifier. Labels with sentence text and id are returned to the main agent. In the experiment different classification algorithms were used, including k-Nearest Neighbors, Linear SVC, and logistic regression.

IV. Experiments

To evaluate the proposed system, the system model was compared with the proposed system in[13]. This done by constructing training data set and two tests set.

A. Tools

Experiments were conducted in InteliJ IDEA which is integrated development environment (IDE) for Java because its maximize developer productivity. For agent system, JADE framework was used as a most contemporary and well-documented agent-based framework.

As mentioned before, Hadoop ecosystem and HBase have been used for BD storage. Stanford CoreNLP library was used for data preprocessing using Apache Spark. LIBLINEAR library is used for classification and evaluation metrics. For doc2vec model, GENSIM library was used.

B. Agent system implementation

Agent abstraction was incorporated in the framework of the system because it makes easier to build extensible distributed systems with a lot of communicating entities. JADE framework [24][25] was used as a most contemporary and well-documented agent-based framework. In addition to agent abstraction, it provides built-in task composition model, peer-to-peer communication, and agent subscription service.

As shown in Fig1, the system consists of following entities: agents, data models, feature extractors, and classifiers. Agents are the ancestors of JADE's agent class and represent independent steps of the knowledge discovery pipeline.

In addition, a Main Agent coordinates pipeline execution and manages the User Interface (UI). Despite of this, other agents can operate independently. For example, one can query knowledge agent for available relations.

Feature extractors and classifiers represent different features and classification algorithms used for relation extraction. Agents communicate via JADE messaging system and JADE yellow pages. There are two types of communication messages, coordination messages, and payload messages. Coordination messages serve to orchestrate user query execution among agents. Payload agents carry data such as sentences or classifiers. Jade yellow pages were used by the Main agent to check and get the list of agents who exist on the system

C. Features extraction

For a bag of word model features, feature-specific information has been extracted from the train sentences in the form of a token set. Then Term Frequency-Invers Document Frequency (TF-IDF) algorithm was applied. If several features were used for classification, resulting feature matrix is obtained by concatenation of feature vectors for both features.

For doc2vec, the recommendations of [23] article was followed. both distributed bag of words and distributed memory variations of the algorithm have been used.

For all the classification algorithms, the default parameters and settings were used.

D. Training set construction

The training set was constructed from sentences that matched MRREL relations with our own method as mentioned in section 3 and inspired by[13]. However, the model differs in two aspects: a semantic type of the entities is used to get all the relations between the biomedical entities in UMLS KB, and we used general relation examples that appear between the given semantic types to construct the negative examples. In contrast authors of [13], used only pairs that participate in "may_treat" relation, regardless of their semantic type.

To enhance the training set quality, we applied filtering by part of speech tag. MetaMap tool has a most common error that is annotating verbs or adjectives as if they were nouns as observed by manual check. Using CoreNLP library as in [20] we annotated each sentence in training set with part of speech

tags and threw away those sentences which concept was not marked as nouns.

For the training set labeling, all relations were divided into two groups: specific relations that labeled with "RO" in MRREL, where RO relation described as has a relationship other than synonymous, narrower, or broader, and other than RO relation groups that represent more general relations. General relations were considered as negative examples for classification and labeled as "other." Sentences with multiple "RO" relations were not included in a training set because they could represent any of those relations, but classifier needs the exact match with label and ground truth. We also discard non-frequent relations.

Another observation was that "RO may_treat" relation almost include"RO=may_prevent" relation and all most of the sentences labeled with "may_prevent" were also labeled with "may_treat". Manual analysis showed that ground truth for such sentence could be either of both relations as shown in example 1 that the treatment "desferrioxamine" treats the" iron overload", and they are indistinguishable by MRREL. We decided to unite such relations into one more general.

Example 1: [Intensified **desferrioxamine** (TREATMENT) treatment (by either subcutaneous or intravenous route) or use of other oral iron chelators, or both, remains the established treatment to reverse cardiac dysfunction due to **iron overload** (DISEASE)]

Since our target examples of relation is "may_treat" we observed that "null" and "related_to" relations will not serve this relation between treatment and disease entities, if we consider example 2, we can observe that "METABOLIC SYNDROME" does not treat or prevent the disease "CHOLESTEROL", but they are related to each other in another way. For that, we exclude "null" and "related_to" examples from the training data set examples.

Example 2: [BACKGROUND: To establish the rate of agreement in predicting **METABOLIC SYNDROME** (TREATMENT) (ms) in different pediatric classifications using percentiles or fixed cut-offs, as well as exploring the influence of **CHOLESTEROL** (DISEASE)]

E. Test set construction

Two data sets have been used to evaluate the performance of the classifier model. The first test set constructed by combining different relation mining data sets so that it could be similar to a training set. The second test set we used the same test set presented in [13] after their permission.

In the first test set, we employed three most specific and frequent relations: "may_treat", "gene_product_malfunction_associated_with_disease " and "other" to serve our training set that contains these relations. Further, we identify this data set as "Triple relation" test set (for simplicity). For this test set, 70 examples of "other" relation were labeled manually. 500 "may_treat" examples and 60 "other" examples were obtained from disease-treatment relations test set in [19]. 500 examples of "gene_product_malfunction_associated_with_disease" were randomly chosen among positive examples of gene-disease relation test set in [26].

The second test set from [13] contains 227 examples of "other" relations and 173 examples of "may_treat" relations. This set is called "may_treat." test set. Since it is important to keep in training set only those relations that presented in the test set, we exclude the relation "gene_malfunction_is_associated_with_disease" from the training set examples to evaluate using the test set "may_treat" from [13].

V. Result and Discussion

Because preprocessing works independently for each sentence, this job is highly parallelizable. We used Spark framework to do the parallelization. Since data was represented as a collection of compressed files, parallel processing was done file-wise. Performance results are summarized in Table. 1 when using a Spark in preprocessing step with CoreNLP for 4000 sentences, which indicate that using spark with a different number of worker reduce the time which means it speed up the processing step.

Different measurements are used to measure the performance of the system. The main purpose of measuring the performance is to compare the system with other systems to determine the success of the proposed design. In the literature, the most widely used evaluation metrics are Accuracy, Precision, Recall, and F-Score. Thus we used these measurements that most common metrics used in classifier evaluation which defined in equations (1), (2), (3) and (4) respectively:

$$Accuracy = \frac{tp+tn}{tp+tn+fn+fp} \quad (1)$$

$$Precision = \frac{tp}{tp+fp} \quad (2)$$

$$Recall = \frac{tp}{tp+fn} \quad (3)$$

$$F - Score = 2 * \frac{precision*recall}{precision+recall} \quad (4)$$

Where (tp) is the true positive results of classification and (fp) is the false positive results of classification and (fn) is the false negative.

On "Triple test set", the values of Precision, Recall and F-Sore has calculated for each class, and then a weighted average is calculated.

TABLE I. PERFORMANCE OF MEDLINE DATA PREPROCESSING USING SPARK FRAMEWORK

Experiment number	1 worker	2 workers	4 workers	No spark
1	239.9	161.1	146.8	434
2	245.9	163.2	144.4	420
3	245.2	162.8	143.8	411
4	241.4	165.4	144.2	420
5	244.9	170.1	143.6	434
Average	243.46	164.52	144.56	423.8

Based on Fig 2, the best result of self-supervised approach in [13], achieved when the baseline data set restricted to 10,000 training instances.

On the system, a different combination of features that discussed in section 3 and different classification algorithm have been applied to evaluate the model on both test sets. Based on "Triple test set", the model shows a better result in terms of Accuracy and Precision when using Linear SVM as the algorithm of classification and Words between entities as basic feature to represent the sentences as shown in Fig. 3 and in term of Recall, and F-Score by using KNN with Euclidean cosine metric with words between entities and words on semantic features. On the same test set and by applying paragraph to vector as an additional feature with the other features as shown in Fig. 4, the best result achieved in Precision, Recall, and F-Score, when using Linear SVM with a paragraph to vector concatenated with words on the semantic path and words between entities features, and in term of Accuracy by using Logistic regression with paragraph to vector concatenated with words on the semantic path and words between entities features.

Furthermore and based on "may_treat" and in comparison with paper[13], the better result as shown in Fig. 5, achieved in the term of Recall and F-Score when using Words between entities features with words on semantic path features using Linear SVM algorithm and in terms of Accuracy and Precision when using Logistic regression with words between entities feature. Fig. 6 shows that the best result after adding a paragraph to vector as an additional feature with the other features is achieved in term of Recall and F-Score by concatenating paragraph vectors with words on the semantic path and words between entities using Linear SVM algorithm. In term of Precision the best result achieved by using KNN with cosine distance metric and paragraph vectors with words between entities and words on semantic path features. The best Accuracy result achieved by applying paragraph to vectors with words on the semantic path using Logistic regression.

The above discussion showed that the system results outperform results from [13]. The reason is that the authors in [13] took sentences that contain random disease-treatment entity pairs which not presented in knowledge base in "may_treat" relation, but due to incompleteness of actual UMLS MRREL knowledge base, those pairs are still very likely to have the target relation "may_treat", so they will have some portion of positive sentence labeled as negative that confuse classifier and harm its performance. On the other hand, in proposed method, only the pairs, which participate in relations other that the target relation, was used to label the negative examples. Those pairs are much less likely to be positive examples, so the train set has higher labeling quality which increases classifier performance. Also, this hypothesis was also confirmed by visual analysis of obtained train sets.

Moreover, by using paragraph vector features, the system results increased as shown in Fig. 4 and Fig. 6, the reason, as justified in [21] and[22], is that in contrast to other features of a bag of the world model, doc2vec captures word semantics that gives additional information to a classifier which enhances the classifier performance.

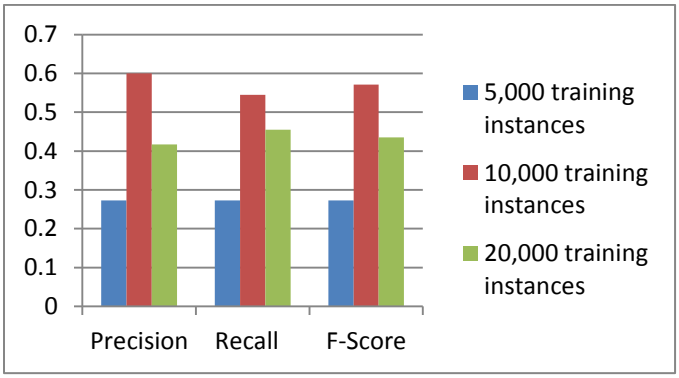

Fig. 2. The result of [14] based on "may_treat" test set

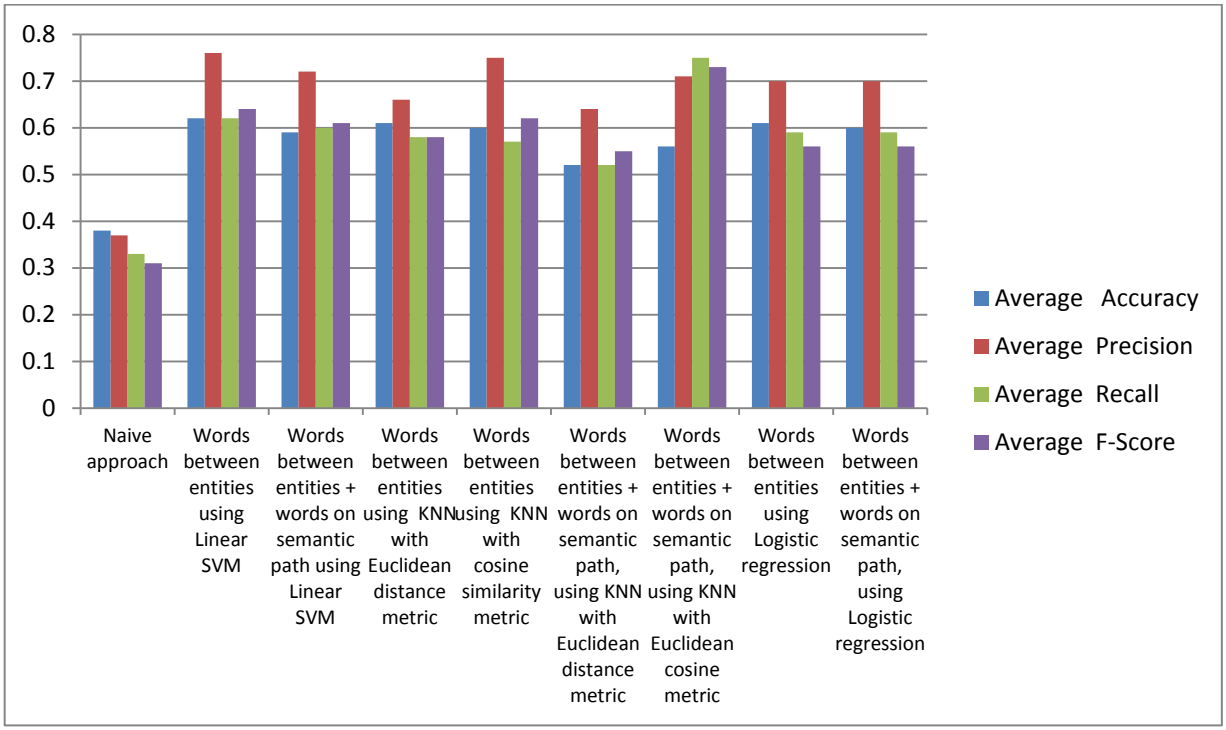

Fig. 3. The result of "Triple test set"

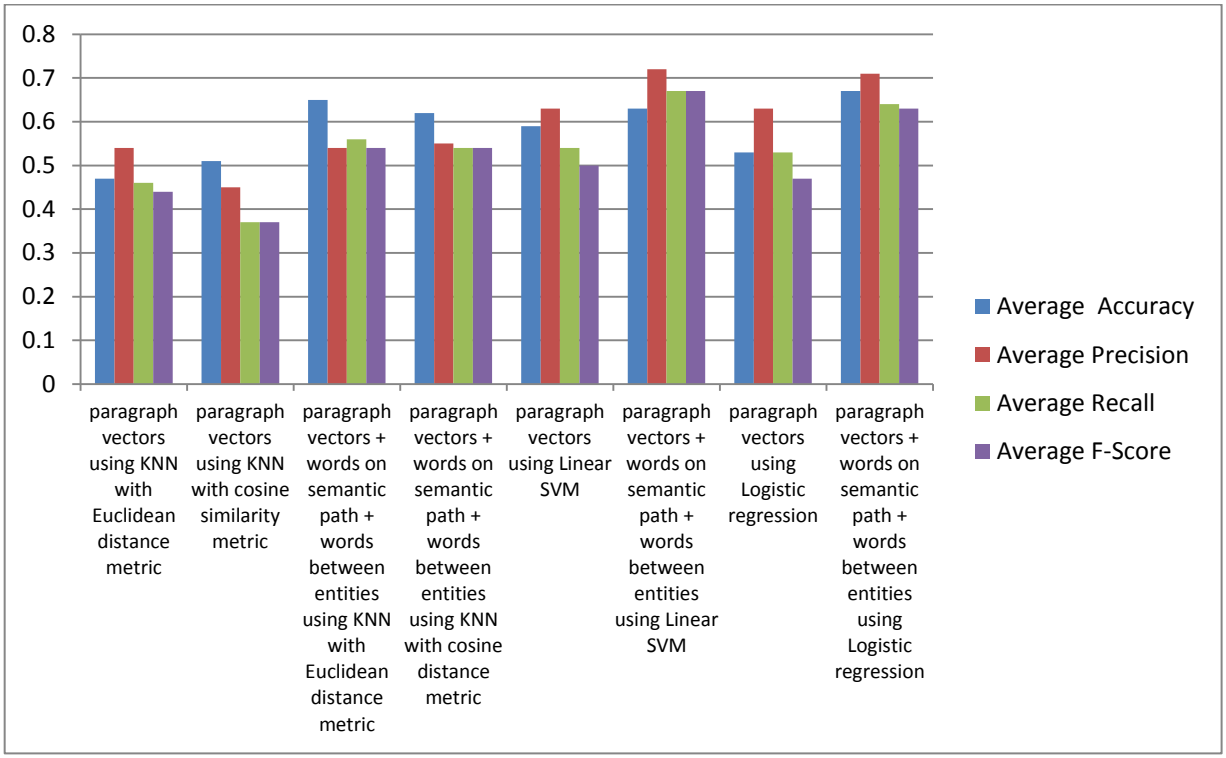

Fig. 4. The result of "Triple test set" with paragrph to vector

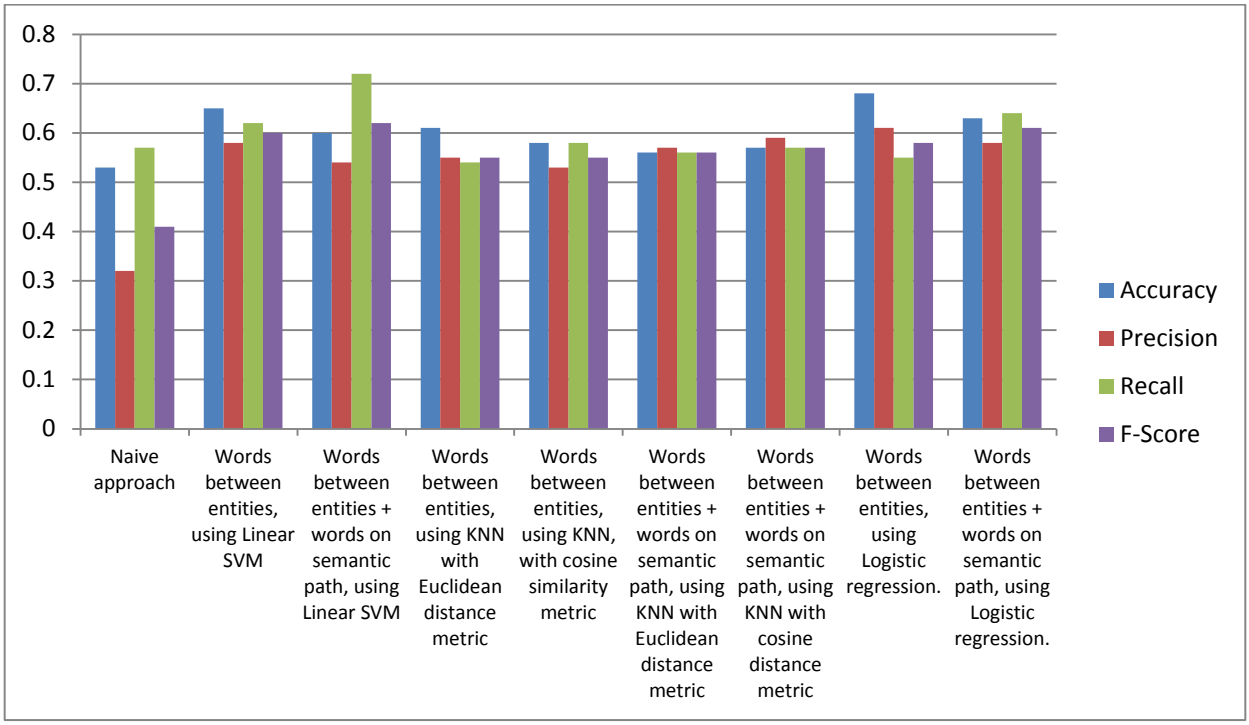

Fig. 5. The result of "may_treat" test set

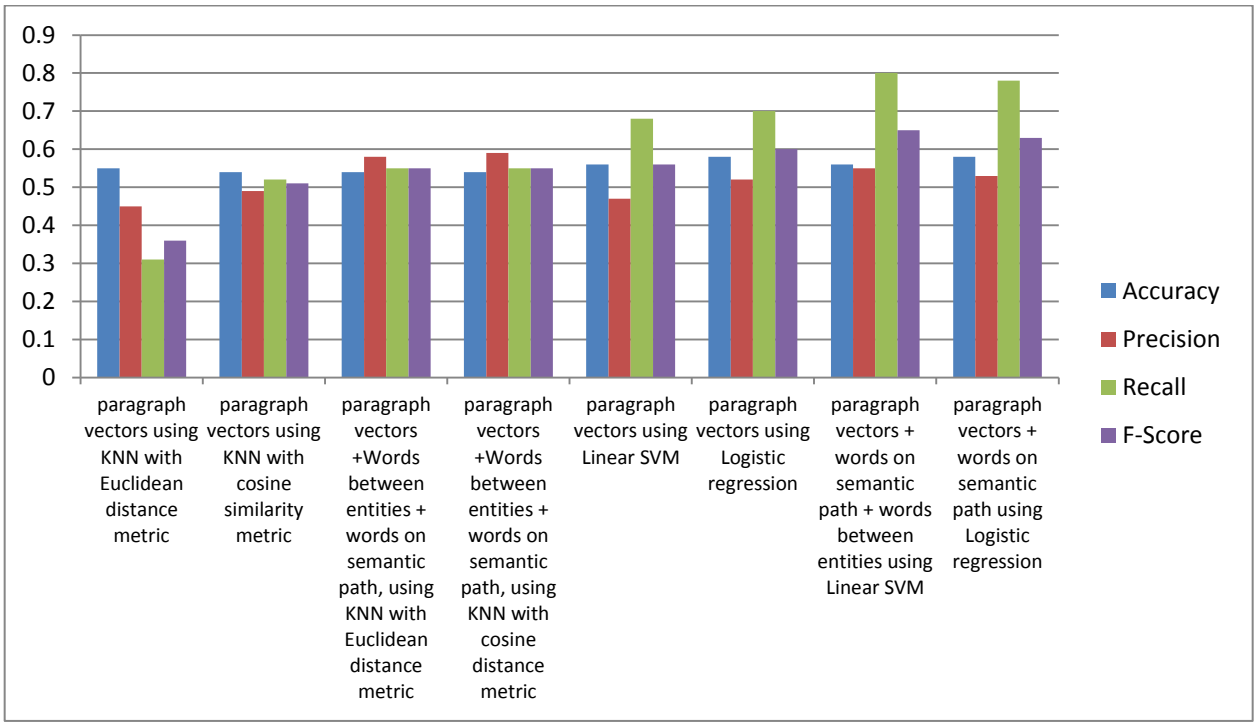

Fig. 6. The result of "may_treat" test set with pargraph to vector

VI. Conclusion

This paper presented a model of big data knowledge mining for Self-supervised relation extraction between biomedical entities from MEDLINE biomedical texts using UMLS knowledge base. The system model fundamentally focused on the extraction of semantic relations between treatments and diseases. The model use hybrid features sets which are: The document to vector, sequence of words between entities, words on the semantic path between entities to enhance the classification performance. The system use a Self-supervised approach for relation extraction by incorporating DM and ML with MAS techniques and demonstrate model performance on MEDLINE data and

UMLS knowledge base to constructing training examples. Moreover, the model used Spark technology with HBase to speeding up the processing of such BD corpus which indicates that using spark with more than two workers will speeding up the preprocessing step. The results also showed that the presented model achieved better results by adopting different features representation and running different classifier algorithms comparing with outperform naïve approach and other paper approach in terms of Accuracy, Precisions, Recall and F-Score. The model also demonstrates an approach to minimize the cost of relation extraction by using a weekly labeled training example using UMLS.

VII. Scope of Future Work

The future work can be classified into two categories, first: improving the performance of relation extraction quality by using Bootstrapping relabeling technique in [27], which can enhance labeling quality. Second improvement is extending train set size with usage of several knowledge bases such as UMLS and IntAct. Using both contain examples of protein-protein interactions. Using both of them can gather more examples and label more data and building manual mapping to unify all the relation representation of each knowledge base or develop sophisticated algorithm that can discover such mapping automatically.

References

[1] M. R. Wigan and R. Clarke, "Big Data's Big Unintended Consequences," *Computer*, vol. 46, pp. 46-53, 2013.

[2] A. Bchir and W. Ben Abdessalem Karaa, "Extraction of drug-disease relations from MEDLINE abstracts," in *Computer and Information Technology (WCCIT), 2013 World Congress on*, 2013, pp.3-1 .

[3] W. Xindong, Z. Xingquan, W. Gong-Qing, and D. Wei, "Data mining with big data," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 26 ,pp. 97-107, 2014.

[4] L. R. Sebastian, S. Babu, and J. J. Kizhakkethottam, "Challenges with big data mining: A review," in *Soft-Computing and Networks Security (ICSNS), 2015 International Conference on*, 2015, pp. 1-4.

[5] O. Rusu, I. Halcu, O. Grigoriu, G. Neculoiu, V .Sandulescu, M. Marinescu, *et al.*, "Converting unstructured and semi-structured data into knowledge," in *Roedunet International Conference (RoEduNet), 2013 11th*, 2013, pp. 1-4.

[6] E. Begoli and J. Horey, "Design Principles for Effective Knowledge Discovery from Big Data," in Software Architecture (WICSA) and European Conference on Software Architecture (ECSA), 2012 Joint Working IEEE/IFIP Conference on, 2012, pp. 215-218.

[7] S. Benomrane, M. Ben Ayed, and A. M. Alimi, "An agent-based Knowledge Discovery from Databases applied in healthcare domain," in *Advanced Logistics and Transport (ICALT), 2013 International Conference on*, 2013, pp. 176-180.

[8] V. N. Romero, S. Kudama, and R. Berlanga Llavori, "Towards the Discovery of Semantic Relations in Large Biomedical Annotated Corpora," in *Database and Expert Systems Applications (DEXA), 2011 22nd International Workshop on*, 2011, pp. 465-469.

[9] Y. Lin, S. Cheng-Jie, W. Xiao-Long, and W. Xuan, "Relationship extraction from biomedical literature using Maximum Entropy based on rich features," in *Machine Learning and Cybernetics (ICMLC), 2010 International Conference on*, 2010, pp. 3358-3361.

[10] L. Mengwen, L. Yuan, A. Yuan, H. Xiaohua, A. Yagoda, and R. Misra, "Relation extraction from biomedical literature with minimal supervision and grouping strategy," in *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on*, 2014, pp. 444-449.

[11] P. Thomas, "Robust relationship extraction in the biomedical domain," Mathematisch-Naturwissenschaftliche Fakultät, 2015.

[12] P. Thomas, I. Solt, R. Klinger, and U. Leser, "Learning protein protein interaction extraction using distant supervision," *Robust Unsupervised and Semi-Supervised Methods in Natural Language Processing*, pp. 34-41, 2011.

[13] R. Roller and M. Stevenson, "Self-supervised Relation Extraction Using UMLS," in *Information Access Evaluation. Multilinguality, Multimodality, and Interaction*. vol. 8685, E. Kanoulas, M. Lupu, P. Clough, M. Sanderson, M. Hall, A. Hanbury, *et al.*, Eds., ed: Springer International Publishing, 2014, pp. 116-127.

[14] R. Roller and M. Stevenson, "Applying UMLS for Distantly Supervised Relation Detection," in *Proceedings of the 5th International Workshop on Health Text Mining and Information Analysis (Louhi)*, 2014, pp. 80-84.

[15] P. Thomas, T. Bobic, M. Hofmann-Apitius, U. Leser, and R. Klinger, "Weakly Labeled Corpora as Silver Standard for Drug-Drug and Protein-Protein Interaction," *Third Workshop on Building and Evaluating Resources for Biomedical Text Mining Workshop Programme*, p. 63, 2012.

[16] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," presented at the Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2, Suntec, Singapore, 2009.

[17] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," presented at the Proceedings of the 2010 European conference on Machine learning and knowledge discovery in databases: Part III, Barcelona, Spain, 2010.

[18] R. Hoffmann, C. Zhang, and D. S. Weld, "Learning 5000 relational extractors," presented at the Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 2010.

[19] B. Rosario and M. A. Hearst, "Classifying semantic relations in bioscience texts," presented at the Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain, 2004.

[20] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, "The Stanford CoreNLP Natural Language Processing Toolkit," in *ACL Demonstrations*, 2014.

[21] Q. V. Le and T. Mikolov, "Distributed Representations of Sentences and Documents," *CoRR*, vol. abs/1405.4053, / 2014.

[22] A. M. Dai, C. Olah, and Q. V. Le, "Document embedding with paragraph vectors," *arXiv preprint arXiv:1507.07998*, 2015.

[23] R. Rehurek and P. Sojka, "Software framework for topic modelling with large corpora," in THE LREC 2010 WORKSHOP ON NEW CHALLENGES FOR NLP FRAMEWORKS, 2010, pp. 45--50.

[24] F. Bergenti, G. Caire, and D. Gotta, "Agents on the move: JADE for Android devices," in *Procs. Workshop From Objects to Agents*, 2014.

[25] J. P. Müller and K. Fischer, "Application Impact of Multi-agent Systems and Technologies: A Survey," in *Agent-Oriented Software Engineering: Reflections on Architectures, Methodologies, Languages, and Frameworks*, O. Shehory and A. Sturm, Eds., ed Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 27-53.

[26] À. Bravo, J. Piñero, N. Queralt-Rosinach, M. Rautschka, and L. I. Furlong, "Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research," *BMC Bioinformatics*, vol. 16, pp. 1-17, 2015

[27] W. Wu, H. Li, H. Wang, and K. Q. Zhu, "Semantic Bootstrapping: A Theoretical Perspective," *IEEE Transactions on Knowledge and Data Engineering*, vol. PP, pp.2016 ,1-1 .