Automatic Relationship Verification in Online Medical Knowledge Base: a Large Scale Study in SemMedDB

Danchen Zhang[1], Daqing He[1], Ning Zou[1], Xin Zhou[1], Fen Pei[2]
[1]School of Computing and Information, [2]School of Medicine
University of Pittsburgh
Pittsburgh, USA
daz45, dah44, niz19, xiz172, fep7@pitt.edu

***Abstract*—Automatically generated public medical knowledge bases (KBs), such as SemMedDB, are commonly used in various medical informatic tasks because of their comprehensive coverage. However, due to the imperfectness of the automatic algorithms for generating those KBs, they often contain noisy statements about medical concepts and relationships. For example, the extraction precision of SemRep, the tool used for constructing SemMedDB, is reported be 74.5%. Previous work focused on improving the algorithms for more accurate extraction. In this paper, however, we propose a supervised learning method to automatically verify the medical relationships. Through a study conducted on SemMedDB, we develop a method for generating a large set of training data with a relative small human labor annotation cost. We further propose nine features to characterize each medical relationship instance. After testing on several classifiers, our proposed methods can achieve the best F1 score and Accuracy at 80%, which demonstrates the effectiveness of our approach. In summary, our study demonstrates that noisy relationships in large scale medical KBs can be identified and removed without much human involvement.**

***Index Terms*—knowledge graph verification, classification**

I. Introduction

With the abilities of clearly stating the meaning of concepts and the relationships among the concepts, knowledge bases (KBs) have been applied in many medical related scenarios, including providing reference knowledge to refine queries in a medical retrieval system [1], helping doctors to make clinical decisions [2], and assisting medical chat bots with medical evidences [3]. A few examples of such medical knowledge bases are presented in Table I, which are usually in the form of *[head concept, relationship, tail concept]*.

There is a trade-off between the quality and the coverage of medical KBs. For example, the Unified Medical Language Systems (UMLS) [4] is a manually generated medical KB with such high quality that it can often be used as a gold standard for many medical knowledge. However, its coverage is relative limited. For example, according to our exploration, after removing obsolete data [1], there are 13.4 million entries in UMLS. However, if we focus on treatments, UMLS only contains treatment information for 1,028 diseases, which is far from sufficient.

In contrast, automatically generated public medical KBs such as Semantic MEDLINE Database (SemMedDB) [5], is usually not as accurate due to the limitations of the extraction algorithms used. For example, the extraction precision of SemRep, the tool used to generate SemMedDB, is only 74.5% [6], and [7] identified many types of extraction errors generated by SemRep. However, the coverage of SemMedDB is much more comprehensive. It contains 18.5 million entries after removing non-novelty data [8], which is huge comparing to UMLS. Taking treatment as an example, SemMedDB contains treatment information for 31,278 diseases.

However, because it contains many noisy information, directly applying SemMedDB sometimes may lead to sub-optimal performance. For example, Zhang et al. [1] found that SemMedDB could provide inaccurate medical knowledge than unstructured Wikipedia in their diagnosis prediction task. Zhang et al. [9] had to manually clean the comparative and treatment relationships from SemMedDB before using them.

Among all the noisy information could exist in SemMedDB, we are particularly interested in identifying potentially wrong relationships. Table I shows two such examples from SemMedDB and corresponding data from UMLS. Between given concept pairs, SemMedDB provides several relationships. It also presents the number of sentences where the relationship is extracted from. We call such number the *support* for that relationship, and it is presented as the right most column in Table I. The first example shows three relationships between *Anti-Inflammatory Agents* and *Aspirin*, and the second example shows five relationships between *Reserpine* and *Hypertensive disease*. In the first example, *Aspirin* actually is an *Anti-Inflammatory Agent*, so all three relationships in Table I are wrong. In the second example, according to UMLS and our expert annotation, *treats, affects* and *disrupts* are correct, *causes* is wrong, while *associated_with* is not sure. In fact, there is a direct contradiction between *treats* and *causes*.

Consequently, to improve the utility of automatically generated medical KBs, in this paper, we target on verifying the relationships in medial KBs. We select SemMedDB for this study because it is probably the most commonly used automatically generated public medical DB, and yet it contains

[1] https://www.ncbi.nlm.nih.gov/books/NBK9684/

TABLE I
EXAMPLES FROM SEMMEDDB AND UMLS

Head Concept	SemMedDB Relationship	Tail Concept	Support
Anti-Inflammatory Agents, Non-Steroidal	COEXISTS_WITH;	Aspirin	2
Anti-Inflammatory Agents, Non-Steroidal	INTERACTS_WITH	Aspirin	4
Anti-Inflammatory Agents, Non-Steroidal	ISA;	Aspirin	574
Reserpine	DISRUPTS	Hypertensive disease	1
Reserpine	CAUSES	Hypertensive disease	1
Reserpine	ASSOCIATED_WITH	Hypertensive disease	1
Reserpine	AFFECTS	Hypertensive disease	4
Reserpine	TREATS	Hypertensive disease	121

Head Concept	UMLS Relationship	Tail Concept	
Anti-Inflammatory Agents, Non-Steroidal	inverse_isa	Aspirin	
Reserpine	may_treat	Hypertensive disease	

lots of noisy relationships due to its huge size [12]–[14], [22]–[24].

In order to obtain high accuracy in identifying and resolving noisy relationships in SemMedDB, we take a supervised learning approach. There are therefore two main challenges to be resolved in this study. The first challenge is to construct a training dataset with adequate labeled data. Because it is too expensive to manually annotate hundreds of thousands medical relationships, our method, therefore, uses the relationships in UMLS as the ground truth for labeling SemMedDB relationships, which enabled us to cheaply and quickly build up a big training set. Due to their size differences, UMLS can only help to label a relatively small part of SemMedDB relationships. Furthermore, these two KBs use different terms for expressing the relationships, so we recruited two experts with medical knowledge to annotate the relationship mappings between two KBs with the help of the relationship definition. The second challenge is how to characterize each type of relationships, that is how to identify significant features to represent the relationships. Later in the methodology section, we will present the details on addressing this challenge.

The contributions of this study are:

(1) The problem explored in this study is innovative. Past studies [10], [11], [15] concentrated on improving knowledge extraction algorithms or combine several KBs to improve the utility. Our focus is on automatically verifying the extracted relationships to improve the KB's utility.

(2) We developed a method to construct a large size of training data for medical relationships without too much human annotation effort, and identified a list of relationship mapping from UMLS to SemMedDB.

(3) We proposed and developed an effective automatic relationship verification model. Through combining both the conceptual and semantic evidences, the model can obtain 80% performance on both Accuracy and F1 measure.

In the reminder of the paper, we will first show the related works in Section 2, then methodologies is demonstrated in Section 3. Experiment and results are described in Section 4. We have more discussion in Section 5, and finally give our conclusion and future work in Section 6.

II. Related works

Knowledge representation, extraction, application has always been the important topics in medical informatics. In this section, we show the related works on medical KB's construction, usage and evaluation.

A. Medical Knowledge Base Construction

Broadly speaking, a knowledge base can be constructed in two ways: manually generated or automatically extracted. UMLS is a commonly used manual generated KB, and some other examples include dbMAE, a KB of autosomal monoallelic expression [25], and Freebase, a general KB collaboratively composed by it community members [27].

KBs can also be constructed using information extraction techniques. MetaMap [21] is an example algorithm for medical concept extraction. Large quantity of text can be processed for constructing comprehensive KBs with much cheaper efforts. Using SemRep, another program for extracting subject-relation-object triples from biomedical free text, SemMedDB is constructed as the largest openly available medical KB, and its coverage and quality has kept improving since 2003 [18]. dRiskKB is a KB constructed from MedLine using a semi-supervised iterative pattern learning approach [16]. Wang et al. [20] reported a KB extracted by a manifold medical relation extraction model.

B. Usage of Medical Knowledge Bases

Due to imperfection of automatic algorithms, a large automatically generated KB could contain various noisy information. There is a need for verifying the relationships in the KB. To obtain precise knowledge from automatically generated KB, manually annotation before direct usage could be applied. For example, Zhang et al. [9] manually cleaned extracted comparative and treatment relationships from SemMedDB before applied them in their clinical decision support tasks. However, manual work is too expansive, or even impossible when faced to a large data.

Another way to obtain more precise knowledge is to combine several KBs. For example, Wei et al. [11] derived information from RxNorm, Side Effect Resource 2, MedlinePlus, and Wikipedia to get accurate medication information. Bejan et al. [10] used the medication information from MEDI and evidence from SemRep to supplement the features needed for identifying treatment relationships. Both studies showed that combining evidence from multiple KBs is more reliable.

However, different KBs may contain different domain knowledge that cannot be easily combined. For example, Bejan et al. [10] found that SemMedDB and MEDI only share 9% of their relationships. Also, different KBs use different kinds of relationships. For instance, SemMedDB, MEDI, and UMLS are using different relationship systems to connect their concepts [4], [5], [11]. Consequently, manually efforts had to be applied in order to map relations among different KBs. For example, Vizenor et al. [17] manually examined concepts and their relationships in order to map UMLS relationships to relationships in Semantic Networks.

C. Medical Knowledge Bases Relationship Verification

Many works try to evaluate a KB by its entropy [26], [28], [29], which is based on the assumption that all relationships in KB are correct, and a more dense KB means more information.

However, Kilicoglu et al. [15] tried to evaluate the factuality of the individual relationships in SemMedDB. They manually annotated 500 PubMed abstracts as the training data, and proposed a compositional approach to extract lexical and syntactic features from the sentences where the individual relationship is extracted from so that the quality of the individual relationship can be evaluated. However, this work does not fit the scenario where the original text is not available (i.e., only have a KB graph). Therefore, we propose in this paper a method that only utilities the KB graph structure to automatically verify the relationships.

III. Methodologies

A. Collecting Labeled Training Data

Building a robust supervised model requires a large amount of training data. Obtaining such amount of training data via manual annotation imposes two challenges. Firstly, it is just too expensive and too slow to do it manually. Secondly, it requires the annotators to have adequate domain knowledge, which in our case medical expertise. Consequently, we seeked to obtain high quality medical knowledge for generating training data via other means.

Our approach is to take the advantage of UMLS, which is a manually generated KB, to generate labeled training data. There are 218,452 medical concept pairs appearing in both UMLS and SemMedDB. *[anti-inflammatory agents, aspirin]* in Table I is such a concept pair appearing in both KBs. We call those shared concept pairs the overlapping data between the two KBs. Because each overlapping medical concept pair usually has different sets of relationships identified in SemMedDB and UMLS, respectively, we can obtain a set of UMLS-SemMedDB relationship pairs based on those two sets of relationships. Again, using the concept pair *[anti-inflammatory agents, aspirin]* in Table I as an example, they can generate relationship pairs such as *[inverse_isa, COEXISTS_WITH]*, *[inverse_isa, ISA]*, and *[inverse_isa, INTERACTS_WITH]*.

Because UMLS was constructed manually, when there is a conflict in a relationship pair, we would usually think that it is the relationship from SemMedDb to be wrong. This is because SemMedDB was generated automatically, whose relationships can have higher chance to be wrong. For example, UMLS states that "may_treat" relationship exists between *[anti-inflammatory agents, aspirin]*, whereas SemMedDB thinks that the relationship ought to be "ISA", due to the conflict meaning between the two relationships, we would think that the relationship in SemMedDB may be wrong. If every relationship pair is checked for such conflict, we would be able to use UMLS data to clearly label the overlap part of SemMedDB data as "correct" relationships or "wrong" relationships. This not only helps to clean SemMedDb data, it also creates the training data for automatically predicting other SemMedDB relationships to be correct or wrong.

However, as stated in Section II-B, it is not straightforward to automatically map relationships in UMLS to those in SemMedDB. See Table I for examples of lacking clarity in the mapping. Consequently, the procedure for collecting labeled training data consists of two steps. The first step is to establish reliable labels on the relationship pairs through expert manual annotation. The second step is then to automatically assign the labels to the relationships in SemMedDB based on the annotated relationship pairs. In the remaining of this section, we will present these two steps in details.

During the first step, based on the overlap concept pairs between UMLS and SemMedDB, we performed automatic identification of relationship pairs. Then, we hired two experts[2] with medical background to manually identify correct relationship pairs from UMLS to SemMedDB. In order to maintain the annotation quality, we provided the annotators with relationship definitions. Our annotators needed to read the relationship definitions, and examined the relationships in the overlapping data in order to assign the following labels:

- The label **correct** is assigned to a relationship pair *[relationship_UMLS, relationship_SemMedDB]* if for any concept pair *[head concept, tail concept]* that satisfies UMLS relationship *[head concept, relationship_UMLS, tail concept]*, it is logical to think that the concept pair also satisfies SemMedDB relationship *[head concept, relationship_SemMedDB, tail concept]*. For example, relationship pair *[isa, ISA]* is labeled as correct.
- The label **wrong** is assigned to a relationship pair *[relationship_UMLS, relationship_SemMedDB]* if for any concept pair *[head concept, tail concept]* that satisfies UMLS relationship *[head concept, relationship_UMLS, tail concept]*, it is logical to think that the concept pair cannot satisfies SemMedDB relationship *[head concept, relationship_SemMedDB, tail concept]*. For example, relationship pair *[inverse_isa, ISA]* is labeled as wrong.
- The label **not sure** is assigned to a relationship pair *[relationship_UMLS, relationship_SemMedDB]* if the anno-

[2] Both annotators obtained bachelor degree in Pharmaceutical Sciences, and master degree in Medicinal Chemistry.

tator cannot find convincing logic inference from UMLS relationship to SemMedDB relationship. For example, relationship pair *[contraindicated_with_disease, CAUSES]* is labeled as not sure.

The definitions of the three labels show that expert annotators need clear understanding of each relationship in the relationship pairs in order to assign the label correctly. Therefore having clear definitions of each relationship is critical for the annotation task. UMLS data come from 37 data sources, and we successfully collected definitions for 116 UMLS relationships from 10 resources, including FMA, CPT, UMD, RXNORM, NDFRT, NDFRT_FDASPL, NDFRT_FMTSME, SNOMEDCT_US, SNOMEDCT_VET and NCI. These are the UMLS relationships we used in the study, all other UMLS relationships were removed due to lack of clear definitions. Kilicoglu, et al. [32] provided definitions for 60 SemMedDb relationships, and these are the SemMedDB relationships we used. Consequently, we selected in total 3,451 relationship pairs from the overlapping data of two KBs, and each relationship in these pairs has a clear definition.

The definitions of the three labels also show that expert annotators need concept pairs to generate evidence for them to make decisions on the labels. Each concept pair combined with a relationship is an instance of the relationship. The more instances we can find in SemMedDB, the more evidence the annotators have to assign the label correctly. Since this is an initial study, we focused on the relationship pairs that appear very often in SemMedDB. Therefore, we set a threshold of 30 instances in the overlapping part. Consequently, any relationship pair whose SemMedDB relationship appears less than 30 times in the overlapping part of SemMedDB will be removed. Through this filtering, only 392 relationship pairs were left for our experts to manually annotate. All the annotations are available at [3]. The weighted Kappa value of the inter annotation agreement is 0.60. The majority disagreement comes from "wrong" vs "not sure". To maintain the quality of the data used in our training, we further removed 139 pairs disagreed by two annotators. This means that only 253 relationship pairs were left because they were agreed by both experts. Finally, only 163 relationship pairs with "correct" or "wrong" were used to generate training data, while 90 pairs marked as "not sure" were not used.

The second step for generating the training data is to propagate the labels on relationship pairs to the relationship instances in SemMedDB. This is because the supervised model built on the training data will predict whether a triple *[head concept, relationship_SemMedDB, tail concept]* is correct or wrong. This propagation was conducted automatically and was straightforward. Again using the examples in Table I for illustration. Since our annotators agreed that relationship pairs *[inverse_isa, ISA], [inverse_isa, COEXISTS_WITH]* and *[inverse_isa, INTERACTS_WITH]* are labeled as "wrong", based on the relationship instance *[Anti-Inflammatory Agents, inverse_isa, Aspirin]* mentioned in Table I, we can automatically assign label "wrong" to relationship instances *[Anti-Inflammatory Agents, ISA, Aspirin]*, *[Anti-Inflammatory Agents, COEXISTS_WITH, Aspirin]*, and *[Anti-Inflammatory Agents, INTERACTS_WITH, Aspirin]*. Through this way, we finally constructed a training dataset with 72,079 relationship instances, in which 51,503 instances are labeled as "correct", and 20,576 instances are labeled as "wrong".

We do notice that, Cui et al. [33] reported that 138,987 concept pairs in UMLS were found to have inconsistent relationships across multiple sources. However, it only takes 1.03% of all concept pairs in UMLS, and also the mapping between UMLS and SemMedDB relationships is manually constructed by the experts. Such inconsistent relationships in UMLS will only have a very limited effect, and is ignored in the current study.

B. Collecting Labeled Testing Data

To objectively evaluate our proposed relationship verification strategy, a testing dataset is needed beyond the training dataset. We adopted a dataset provided by Kilicoglu, et al. [32]. They extracted 1,371 instances from 500 PubMed article abstracts, and manually labeled those instances, which includes 95 instances labeled as "true" because they were correctly extracted from the articles. These 95 instances were checked to see whether or not they appear in the training data. After removing those instances appearing in the training data, we obtained 90 instances as "correct" testing samples.

However, we could not treat the "false" instances as "wrong" testing samples. This is because Kilicoglu, et al. [32]'s method could label an instance to be "false" either because it is a true false instance or because the instance was not correctly extracted.

Since it is hard for us to locate the evidence of "wrong" instances from PubMed papers, our selection of "wrong" instances in the testing collection has to come back to the overlapping data we collected between UMLS and SemMedDB. Based on the annotation presented in Section III-A, among all the relationship pairs marked as either "not sure" or "wrong" but are disagreed by the two annotators, we randomly selected 107 instances with such relationship pairs for another round of manual annotation. Two annotators labeled the correctness of the SemMedDB instances by the evidence from UMLS. For example, UMLS has [enterovirus, causeative_agent_of, enterovirus meningitis], while SemMedDB has [enterovirus, INTERACTS_WITH, enterovirus meningitis], and both annotators labeled this SemMedDB relationship as wrong, because "INTERACTS_WITH" is defined as "substance interaction". Through this round of annotation, we obtained 103 instances marked as "wrong" by both annotators, out of which 90 instances were randomly selected to match the number of "correct" instances. Finally, our testing collection consists of 90 "correct" instances and 90 "wrong" instances.

[3] https://github.com/daz45/SemMedDB-relationship-verification-annotation-data

C. Constructing Weighted Semantic Network

SemMedDB organizes its concepts into hierarchy. For example, *Reserpine* belongs to semantic type “Organic Chemical”, which in turn belongs to Semantic Group “Chemicals & Drugs”. The structural information provided in the hierarchy shows a semantic network [30].

Semantic type information has been an important feature for identifying the relationship between two concepts in several existing automatic KB extraction studies [15], [18], [20]. This triggered us to utilize the semantic type and semantic group information in our model.

The semantic network we constructed from SemMedDB is a weighted network. Each pair of semantic type in the network may have several edges because of different relationships between them, and the weight is the count of the PubMed papers that the relationship appears. Our assumption is that a frequently appearing semantic type pair with a relation has a higher chance to be correct. We applied the same method to construct a network for semantic group too.

D. Identifying Features for Representing Instances

To perform classification for verifying relationships, each relationship instance is represented as a set of features. The verification of a relationship depends on the amount of supports it gets, which can be in the form of the counts of sentences showing the relationship, or the proportion of sentences supporting this relationship against those supporting other relationships, etc.

In total, we identified 9 features which are organized into three groups: concept level, semantic type level, and semantic group level features.

At the **Concept level**, we identified three features:

- **C_support**: the logarithm value of the count of supporting sentences for the relationship between two given concepts. The reason to use the logarithm value rather than the original value is to keep the scale of this feature’s value similar to that of the following two features.
- **C_percentage**:the percentage of the supporting sentences for this relationship among all sentences with the two given concepts.
- **C_isMax**: the sign value to the statement of whether the given relationship has the most number of supporting sentences among all the relationships related to the two given concepts. 1 means yes, and 0 means no.

At the **Semantic type level:**, we identify three features:

- **ST_support**: the logarithm value of the count of supporting sentences for the relationship between two semantic types of the given two concepts.
- **ST_percentage**: the percentage of the supporting sentences for this relationship among all sentences with two given semantic types.
- **ST_isMax**: the sign value to the statement of whether the given relationship between two semantic types has the most supporting sentences all the relationships related to the two given concepts. 1 means yes, and 0 means no.

At the **Semantic group level,** we identify three features:

- **SG_support**: the logarithm value of the count of supporting sentences for the relationship between two semantic groups of the two given concepts.
- **SG_percentage**: the percentage of sentences that supporting this relationship among all sentences with two given semantic groups.
- **SG_isMax**: the sign value to the statement of whether the given relationship between two semantic groups has the most supporting sentences all the relationships related to the two given concepts. 1 means yes, and 0 means no.

E. Classification Methods

As mentioned above, each KB triple consists of two concepts and a relationship between them. The task is to classify whether the relationship between the concepts is correct or wrong. The baseline in this study is c_isMax, which essentially takes the most frequently supported relationship as the correct one. We then selected five classification methods: Gaussian Naive Bayes, Logistic Regression, Random Forest, Decision Tree and k Nearest Neighbor (kNN) for the experiments. Although Supported Vector Machine (SVM) and Neural Network are popular and effective classification methods, they are not selected due to our large training data size and the limited computation resources. All classifiers are based on the implementation in Scikit-learn platform [19].

IV. Experiments and results

A. Experiment Settings and Evaluation Metrics

In the training procedure, 10 fold cross validation was conducted to train the classification models. 90% data was used for training and 10% was used for validation. For each learned model, testing is conducted on 180 testing instances. For each run, cross validation test was repeated for 50 times, and reported scores are average of 50 rounds. The data split is the same across different classifiers, and Wilcoxon significance test was utilized to statistic significance tests. Massive parameters are tried to maximum the classification performance on the training data, and we find most classifiers can achieve best performance with default Scikit-learn parameters except kNN achieve best on training data with k=3.

To evaluate the performance of verifying relationship in SemMedDB, we select the following metrics: $Precision = \frac{|TP|}{|TP|+|FP|}$, $Recall = \frac{|TP|}{|TP|+|FN|}$, $F1 = \frac{2 \cdot Recall \cdot Precision}{Recall+Precision}$, and $Accuracy = \frac{|TP|+|TN|}{|TP|+|TN|+|FP|+|FN|}$, where TP, FP, TN, FN represent true positive (correctly classified as “correct”), false positive (wrongly classified as “correct”), true negative (correctly classified as “wrong”), and false negative (wrongly classified as “wrong”).

B. Relationship Verification Performance

Table II shows the classification performance of the five models and the Baseline on both training and testing datasets. Wilcoxon significance test is conducted on 50 repetitive runs on each run, and numbers labeled with * indicates a significant

TABLE II
CLASSIFICATION RESULTS ON TRAINING AND TESTING DATA

	Training				Testing			
Classifier	**Precision**	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
Baseline	87.44%	92.70%	89.99%	85.27%	65.87%	92.22%	76.85%	72.22%
Gaussian Naive Bayes	88.75%*	90.45%	89.59%	84.99%	80.82%*	79.58%	80.19%*	80.34%*
Logistic Regression	88.27%*	92.65%	90.41%*	85.95%*	70.33%*	90.00%	78.96%*	76.01%*
Decision Tree	96.72%*	97.66%*	97.18%*	95.96%*	52.09%	61.80%	56.49%	52.34%
Random Forest	96.81%*	98.33%*	97.56%*	96.49%*	67.79%*	62.62%	65.09%	66.42%
k Nearest Neighbor	95.68%*	96.96%*	96.32%*	94.70%*	67.74%*	66.67%	67.19%	67.44%

TABLE III
FEATURE LEVEL ABLATION EXPERIMENTS FOR RELATIONSHIP VERIFICATION WITH GAUSSIAN NAIVE BAYES

	Training		Testing	
	F1	**Accuracy**	**F1**	**Accuracy**
Baseline	89.99%	85.27%	76.85%	72.22%
all features	89.59%	84.99%	80.19%	80.34%
C_*	90.01%	85.31%	76.85%	72.22%
ST_*	80.28%	71.87%	74.16%	74.44%
SG_*	84.30%	74.28%	78.02%	77.78%
all - C_*	81.84%	73.68%	66.02%	69.00%
all - ST_*	89.84%	85.22%	80.09%	79.56%
all - SG_*	89.68%	84.98%	80.32%	77.89%

TABLE IV
FEATURE LEVEL ABLATION EXPERIMENTS FOR RELATIONSHIP VERIFICATION WITH DECISION TREE

	Training		Testing	
	F1	**Accuracy**	**F1**	**Accuracy**
Baseline	89.99%	85.27%	76.85%	72.22%
all features	97.18%	95.96%	56.49%	52.34%
C_*	90.00%	85.21%	76.61%	71.50%
ST_*	97.35%	96.17%	54.66%	56.78%
SG_*	97.37%	96.20%	59.15%	59.33%
all - C_*	97.57%	96.49%	60.81%	61.33%
all - ST_*	97.33%	96.16%	61.29%	61.33%
all - SG_*	96.38%	94.80%	57.53%	59.56%

improvement compared with Baseline (p-value$<$0.01). Baseline has a Precision at 87.44%, but Recall at 92.70%, indicating that this method generates a lot false positive instances leading to high recall and low precision. This situation is even worse on the testing data. It could imply that some wrong SemMedDB relationships still have quite frequent instances. Further study is needed.

We can also see that, on both training and testing data, Logistic Regression significantly outperforms the Baseline on all metrics except Recall. This shows the utility of the proposed nine features. Furthermore, Gaussian Naive Bayes, although has a slightly worse performance than the Baseline on the training data, performs significantly better than the baseline on the testing dataset. Its performance on the testing data is even better than Logistic Regression. On the other hand, in the training dataset, Decision Tree, Random Forest, and kNN all have 96+% F1 and 94+% Accuracy performance, significantly better than the Baseline. However, their performance are significantly lower in the testing data, indicating a severe overfitting problem. The reasons are further explored in section IV-D.

C. Effectiveness of Features in Each Level

To obtain a better insight on the effectiveness of each level features in our relationship verification task, we performed several feature level ablation experiments using Gaussian Naive Bayes model. As shown in Table III, all - ST_* and all - SG_* give quite similar Accuracy and F1 score, implying that the combination of concept level features with either semantic type or semantic group level features can provide similar information for the relationship verification task. Since Accuracy of all - SG_* is 1.6% less than that of all - ST_*, and SG_* outperforms ST_*, it seems that semantic group maybe a better representation than semantic type for the concepts in this task. However, using both (i.e., all features) still give the most robust performance on training and testing data.

Then, the results show that removing concept level features (i.e., all - C_*) makes the performance drop rapidly, but still all features outperforms C_* by a large degree, indicating that concept level features are essential and play a complementary role to the semantic network features. Also, we do notice that using only concept level features (i.e., C_*) gives the same performance as the baseline, indicating that *C_isMax* is the key feature among three concept level features to Guassian Naive Bayes model.

D. Overfitting of Decision Tree and Random Forest

Different from Gaussian Naive Bayes, Decision Tree and Random Forest have a totally different classification pattern. They both have a very high F1 and Accuracy (95+%) on the training data, but have a very bad performance on the testing data (see Table II). We show the feature level ablation experiments with Decision Tree in Table IV. On the training data, the concept level features gives the worst performance, while the semantic type and group features can achieve 96+% F1 and Accuracy, implying that the classifier can perform well with either semantic type or group features, which is inconsistent to the findings in Table III with Gaussian Naive Bayes. After examining the results on the training data, we find that the instances with same semantic type and semantic group features have been classified to same category. For example, in SG_*, instances with *[Chemicals & Drugs, TREATS, Disorders]* are all classified as "correct", and achieve 96.20% Accuracy on the training data. The reason is because the training data has 5,289 "correct" instances for relationship pair *[may_treat, TREATS]*, and 623 "wrong" instances for relationship pair

TABLE V
MEAN VALUE OF EACH FEATURE IN TWO CATEGORIES

Features	Wrong	Correct
C_support	0.89	1.68
C_percentage	0.34	0.85
C_isMax	0.33	0.93
ST_support	10.10	10.84
ST_percentage	0.31	0.53
ST_isMax	0.36	0.63
SG_support	13.18	13.69
SG_percentage	0.26	0.38
SG_isMax	0.50	0.38

[contraindicated_with_disease, TREATS]. However, since semantic group pair *[Chemicals Drugs, Disorders]* only has 5,794 instances, and majority are "correct" instances, Decision Tree simply classifies all instances with *[Chemicals Drugs, TREATS, Disorders]* as "correct". It learns complicated hyperplanes to fragment the space to fit the training data, but such classifying pattern is not true in dealing with specific instances, and hence had a very bad performance on testing data. We can conclude that Decision Tree and Random Forest are overfitting on the training data.

kNN can achieve 78.76% F1 and 77.22% Accuracy when k is increased to 3,500 on the testing data, implying involving more neighbors can improve model robustness. Table II shows that kNN is like Decision Tree method, performs much worse on the testing data but very well on the training data.

V. Discussions

We find that the Gaussian Naive Bayes and Logistic Regression have similar classification patterns, classifying instances with higher values in concept, semantic type and semantic group level features as "correct", and classifying instances with smaller features values as "wrong". It is consistent with the data distribution of training data. The mean value of each feature in two categories from the whole training data are shown in Table V, and we can see that "wrong" categories truly have a smaller support in 9 features than "correct" categories, except *SG_isMax*. It implies that, in most cases, verifying the relationship according to the proposed 9 features has high utility, and comparing to "wrong" instances, "correct" instances are more frequently extracted from PubMed papers and usually have a higher weight in the semantic network.

Then we examined the instances that the model predicted wrongly in the testing data. There are totally 18 kinds of relationships appearing in the testing data, and relationship verification performance with Gaussian Naive Bayes model on each relationship is shown in Table VI. Among all the relationships, we found that the majority classification errors comes from "TREATS" and "PART_OF". There are 12 "wrong" instances with relationship "TREATS", whose head concepts are all surgery, such as "Operative Surgical Procedures" and "Hysterectomy", and tail concepts are all post surgery diseases, such as "Postoperative fistula", and "Surgical Wound Infection". UMLS gives a "causative_agent_of", implying the surgery cause these post surgery diseases, but SemMedDB gives "TREATS", and both annotators gives "wrong" labels on them. These surgery concepts belong to semantic group "Procedures", and the disease concepts belong to "Disorders". In addition, there is a very high weight for *[Procedures, TREATS, Disorders]* in the semantic network. 11 out of 12 instances have high *c_support, C_percentage* and *C_isMax=1*, and they are mistakenly classified as "correct". The only one that are correctly classified as "wrong" has low *c_support, C_percentage*, and *C_isMax=0*. This implies that wrong instances with smaller values in three level features are easy to be cleaned, but the ones with high support may need extra knowledge or expertise to be recognized. Also, it shows that some errors are repeated many times in the automatic extraction process with SemRep, so that we cannot use frequency counts to identify them.

On the other hand, 8 out of 20 "correct" instances with relationship "PART_OF" are wrongly classified as "wrong", and we find they have very small values on semantic type and group level features. It indicates that such kind of unpopular correct instances are also hard to be recognized by proposed features, simply because only a very small number of PubMed sentences that mentioned such knowledge. Relationship specified features maybe useful to solve this problem, for example if we have prior knowledge that SemRep is very accurate in extracting some particular relationships, we will classify them as "correct" even with less support in current 9 features.

We can summarize that the proposed features are effective in most cases. However, high weight in semantic network only reflects the evidence supporting the relationship between the semantic type or group pairs, but maybe not supportive to the individual instances. For example, we know drug treats diseases, but we do not know whether a specific drug can treat a specific disease or not. Also, it is hard to detect unpopular correct instances.

VI. Conclusions

In this study, we aim to verify the relationships in the automatically generated KB - SemMedDB. We identified nine features in three groups to characterize each medical relationship instance, and proposed a supervised learning model to automatically verify the medical relationships. We tested on several classifiers, and the best one achieved both Accuracy and F1 score 80%, which demonstrates the effectiveness of our proposed methods. Also our analysis shows features related to out extracted weighted semantic network is very effective. Our study indicates that noisy relationships in large scale medical knowledge bases, such as SemMedDB, can be identified and removed with a few of manual annotation workload.

Base on our analysis, we find current features still have limitation, which consider the information from the whole SemMedDB. In the future, more hypothesis and features will be proposed. For example, knowledge extracted from new PubMed papers maybe have higher score than those from old PubMed papers. Subsequently,the features can apply to multiple medical knowledge bases to help identify the noisy relationships without much human involvement.

TABLE VI
VERIFICATION PERFORMANCE OF 18 RELATIONSHIPS IN TESTING DATA

Relationships	Correct count	Wrong count	F1	Accuracy
INTERACTS_WITH	1	42	100%	100%
PROCESS_OF	8	0	100%	100%
AFFECTS	5	0	100%	100%
PREDISPOSES	3	0	100%	100%
DIAGNOSES	2	0	100%	100%
ADMINISTERED_TO	2	0	100%	100%
DISRUPTS	1	0	100%	100%
PREVENTS	1	14	48.28%	93.33%
LOCATION_OF	0	16	44.83%	81.25%
COEXISTS_WITH	4	0	42.86%	75.00%
TREATS	35	12	47.28%	70.21%
CAUSES	3	0	40.00%	66.67%
PART_OF	20	0	37.50%	60.00%
ISA	0	6	33.33%	50.00%
PRODUCES	2	0	33.33%	50.00%
ASSOCIATED_WITH	1	0	0.00%	0.00%
NEG_TREATS	1	0	0.00%	0.00%
NEG_PART_OF	1	0	0.00%	0.00%
total count	90	90		

REFERENCES

[1] Zhang D, He D. Enhancing Clinical Decision Support Systems with Public Knowledge Bases. Data and Information Management.;1(1):49-60.

[2] Berner ES, La Lande TJ. Overview of clinical decision support systems. InClinical decision support systems 2016 (pp. 1-17). Springer, Cham.

[3] Edwards BI, Muniru IO, Cheok AD. Robots to the Rescue: A Review of Studies on Differential Medical Diagnosis Employing Ontology-Based Chat Bot Technology.

[4] Bodenreider O. The unified medical language system (UMLS): integrating biomedical terminology. Nucleic acids research. 2004 Jan 1;32(suppl_1):D267-70.

[5] Kilicoglu H, Shin D, Fiszman M, Rosemblat G, Rindflesch TC. SemMedDB: a PubMed-scale repository of biomedical semantic predications. Bioinformatics. 2012 Oct 8;28(23):3158-60.

[6] Kilicoglu H, Fiszman M, Rosemblat G, Marimpietri S, Rindflesch TC. Arguments of nominals in semantic interpretation of biomedical text. InProceedings of the 2010 workshop on biomedical natural language processing 2010 Jul 15 (pp. 46-54). Association for Computational Linguistics.

[7] Liu Y, Bill R, Fiszman M, Rindflesch T, Pedersen T, Melton GB, Pakhomov SV. Using SemRep to label semantic relations extracted from clinical text. InAMIA annual symposium proceedings 2012 (Vol. 2012, p. 587). American Medical Informatics Association.

[8] Wang L, Del Fiol G, Bray BE, Haug PJ. Generating disease-pertinent treatment vocabularies from MEDLINE citations. Journal of biomedical informatics. 2017 Jan 1;65:46-57.

[9] Zhang M, Del Fiol G, Grout RW, Jonnalagadda S, Medlin Jr R, Mishra R, Weir C, Liu H, Mostafa J, Fiszman M. Automatic identification of comparative effectiveness research from Medline citations to support clinicians treatment information needs. Studies in health technology and informatics. 2013;192:846.

[10] Bejan CA, Denny JC. Learning to identify treatment relations in clinical text. InAMIA Annual Symposium Proceedings 2014 (Vol. 2014, p. 282). American Medical Informatics Association.

[11] Wei WQ, Cronin RM, Xu H, Lasko TA, Bastarache L, Denny JC. Development and evaluation of an ensemble resource linking medications to their indications. Journal of the American Medical Informatics Association. 2013 Apr 10;20(5):954-61.

[12] Ayvaz S, Horn J, Hassanzadeh O, Zhu Q, Stan J, Tatonetti NP, Vilar S, Brochhausen M, Samwald M, Rastegar-Mojarad M, Dumontier M. Toward a complete dataset of drugdrug interaction information from publicly available sources. Journal of biomedical informatics. 2015 Jun 1;55:206-17.

[13] Widdows D, Cohen T. Reasoning with vectors: A continuous model for fast robust inference. Logic Journal of the IGPL. 2014 Nov 18;23(2):141-73.

[14] Cairelli MJ, Miller CM, Fiszman M, Workman TE, Rindflesch TC. Semantic MEDLINE for discovery browsing: using semantic predications and the literature-based discovery paradigm to elucidate a mechanism for the obesity paradox. InAMIA Annual Symposium Proceedings 2013 (Vol. 2013, p. 164). American Medical Informatics Association.

[15] Kilicoglu H, Rosemblat G, Rindflesch TC. Assigning factuality values to semantic relations extracted from biomedical research literature. PloS one. 2017 Jul 5;12(7):e0179926.

[16] Xu R, Li L, Wang Q. dRiskKB: a large-scale disease-disease risk relationship knowledge base constructed from biomedical text. BMC bioinformatics. 2014 Dec;15(1):105.

[17] Vizenor LT, Bodenreider O, McCray AT. Auditing associative relations across two knowledge sources. Journal of biomedical informatics. 2009 Jun 1;42(3):426-39.

[18] Rindflesch TC, Fiszman M. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text. Journal of biomedical informatics. 2003 Dec 1;36(6):462-77.

[19] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J. Scikit-learn: Machine learning in Python. Journal of machine learning research. 2011;12(Oct):2825-30.

[20] Wang C, Fan J. Medical relation extraction with manifold models. InProceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) 2014 (Vol. 1, pp. 828-838).

[21] Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. InProceedings of the AMIA Symposium 2001 (p. 17). American Medical Informatics Association.

[22] Zhang R, Cairelli MJ, Fiszman M, Rosemblat G, Kilicoglu H, Rindflesch TC, Pakhomov SV, Melton GB. Using semantic predications to uncover drugdrug interactions in clinical data. Journal of biomedical informatics. 2014 Jun 30;49:134-47.

[23] Cameron D, Kavuluru R, Rindflesch TC, Sheth AP, Thirunarayan K, Bodenreider O. Context-driven automatic subgraph creation for literature-based discovery. Journal of biomedical informatics. 2015 Apr 1;54:141-57.

[24] Chen G, Cairelli MJ, Kilicoglu H, Shin D, Rindflesch TC. Augmenting microarray data with literature-based knowledge to enhance gene regulatory network inference. PLoS computational biology. 2014 Jun 12;10(6):e1003666.

[25] Savova V, Patsenker J, Vigneau S, Gimelbrant AA. dbMAE: the database of autosomal monoallelic expression. Nucleic acids research. 2015 Oct 25;44(D1):D753-6.

[26] Hempelmann CF, Sakoglu U, Gurupur VP, Jampana S. An entropy-based evaluation method for knowledge bases of medical information systems. Expert Systems with Applications. 2016 Mar 15;46:262-73.

[27] Bollacker K, Evans C, Paritosh P, Sturge T, Taylor J. Freebase: a collaboratively created graph database for structuring human knowledge. InProceedings of the 2008 ACM SIGMOD international conference on Management of data 2008 Jun 9 (pp. 1247-1250). AcM.

[28] Gurupur VP, Sakoglu U, Jain GP, Tanik UJ. Semantic requirements sharing approach to develop software systems using concept maps and information entropy: A Personal Health Information System example. Advances in Engineering Software. 2014 Apr 1;70:25-35.

[29] Doran P, Tamma V, Palmisano I, Payne TR, Iannone L. Evaluating ontology modules using an entropy inspired metric. InWeb Intelligence and Intelligent Agent Technology, 2008. WI-IAT'08. IEEE/WIC/ACM International Conference on 2008 Dec 9 (Vol. 1, pp. 918-922). IEEE.

[30] McCray AT. The UMLS Semantic Network. InProceedings. Symposium on Computer Applications in Medical Care 1989 Nov (pp. 503-507). American Medical Informatics Association.

[31] Zhang D, He D, Zhao S, Li L. Enhancing Automatic ICD-9-CM Code Assignment for Medical Texts with PubMed. BioNLP 2017. 2017:263-71.

[32] Kilicoglu, Halil, et al. Constructing a semantic predication gold standard from the biomedical literature. BMC bioinformatics 12.1 (2011): 486.

[33] Cui, Licong. ”Cohere: Cross-ontology hierarchical relation examination for ontology quality assurance.” AMIA Annual Symposium Proceedings. Vol. 2015. American Medical Informatics Association, 2015.