Scalable, Semi-Supervised Extraction of Structured Information from Scientific Literature

Kritika Agrawal, Aakash Mittal, Vikram Pudi
Data Sciences and Analytics Center, Kohli Center on Intelligent Systems
IIIT, Hyderabad, India
{kritika.agrawal@research.,aakash.mittal@students.,vikram@}iiit.ac.in

Abstract

As scientific communities grow and evolve, there is a high demand for improved methods for finding relevant papers, comparing papers on similar topics and studying trends in the research community. All these tasks involve the common problem of extracting structured information from scientific articles. In this paper, we propose a novel, scalable, semi-supervised method for extracting relevant structured information from the vast available raw scientific literature. We extract the fundamental concepts of *aim*, *method* and *result* from scientific articles and use them to construct a knowledge graph. Our algorithm makes use of domain-based word embedding and the bootstrap framework. Our experiments show the domain independence of our algorithm and that our system achieves precision and recall comparable to the state of the art. We also show the research trends of two distinct communities - computational linguistics and computer vision.

1 Introduction

With the tremendous amount of research publications available online, there is an increasing demand to automatically process this information to facilitate easy navigation through this enormous literature for researchers. Whenever researchers start working on a problem, they are interested to know if the problem has been solved previously, methods used to solve this problem, the importance of the problem and the applications of that problem. This leads to the requirement of finding automatic ways of extracting such structured information from the vast available raw scientific literature which can help summarize the research paper as well as the research community and can help in finding relevant papers. Organizing scientific information into structured knowledge bases requires information extraction (IE) about scientific entities and their relationships. However, the challenges associated with scientific information extraction are greater than for a general domain. General methods of information extraction cannot be applied to research papers due to their semi-structured nature and also the new and unique terminologies used in them. Secondly, annotation of scientific text requires domain expertise which makes annotation costly and limits resources.

There is a considerable amount of previous and ongoing work in this direction, starting from keyword extraction (Kim et al., 2010) (Gollapalli and Caragea, 2014) and textual summarization (Jaidka et al., 2018). Other research has focused on unsupervised approaches such as bootstrapping (Tsai et al., 2013)(Gupta and Manning, 2011), where they introduced hand-designed templates to extract scientific keyphrases and categorize them into different concepts, and then more templates are added automatically through bootstrapping. Hand-designed templates limit their generalization to all the different domains present within the scientific literature. A recent challenge on Scientific Information Extraction (ScienceIE) (Augenstein et al., 2017) provided a dataset consisting of 500 scientific paragraphs with keyphrase annotations for three categories: TASK, PROCESS, MATERIAL across three scientific domains, Computer Science, Material Science, and Physics. This invited many supervised and semi-supervised techniques in this field. Although all these techniques can help extract important concepts of a research paper in a particular domain, we need more general and scalable methods which can summarize the complete research community.

In this work, we propose a new technique to extract key concepts from the research publications. Our main insight is that a paper cites another paper either for its aim, or method, or result. Therefore, key contribution of paper in the research community can be best summarized by its aim, the method used to solve the problem and

Proceedings of the Workshop on Extracting Structured Knowledge from Scientific Publications, pages 11–20
Minneapolis, USA, June 6, 2019.

the final result. We define these concepts as:
Aim: Target or primary focus of the paper.
Method: Techniques used to achieve the aim.
Result: well-defined output of the experiments or contribution which can be directly used by the research community.
Example: "The support-vector network (*Result*) is a new learning machine for two-group classification (*Aim*) problems. The machine conceptually implements the following idea: input vectors are non-linearly mapped to a very high-dimension feature space (*Method*). In this feature space, a linear decision surface is constructed."

We extract these concepts from Title, Abstract and Citation Contexts of a research paper. These sections can be accurately automatically extracted from research papers. Title and Abstract work as a short and to the point summary of work done in the paper. They are an essential place to find the exact phrases for these concepts without the introduction of too much noise. Citation context is the text around the citation marker. This text serves as "micro summaries" of a cited paper and phrases in this text are important candidates for aim, method or result of the cited paper. We combine data mining and natural language techniques to solve the problem scalably in a semi-supervised manner.
Graph representations like knowledge graph that link the information of a large body of publications can reveal patterns and lead to the discovery of new information that would not be apparent from the analysis of just one publication. Analysis on top of these representations can lead to new scientific insights and discovery of trends in a research area. They can also facilitate some other tasks like assigning reviewers, recommending relevant papers or improving scientific search engines. Therefore, we propose to build graphical representation by extracting phrases representing the concepts *Aim*, *Method* and *Result* from scientific publications. We introduce these phrases as additional nodes and connect them to their corresponding paper nodes in the citation graph. We argue that the citation network is an integral part of scientific knowledge graph and the proposed representation can adequately summarize the research community. Proposed graph is shown in Figure 1.

Contributions: Our key contributions are:
(i) We propose a novel, scalable, semi-supervised and domain-independent method for extracting concepts, *aim*, *method* and *result* from the vast available raw scientific literature by using domain-based word embeddings and data mining techniques. Our approach also takes Citation Context into account apart from Title and Abstract on which most of the work relied till now. (ii) We experimentally validate our approach and show statistically significant improvements over existing state-of-the-art models. (iii) We show how the extracted concepts and the available citation graph can be used to represent the research community as a knowledge graph. (iv) We demonstrate our method on a large multi-domain dataset built with the help of DBLP citation network. Our dataset consists of 332,793 papers and 1,508,560 links between them. (v) We present a case study on the computational linguistics community and computer vision community using the three concepts extracted from its articles, for verifying the results of our system and for showing domain independence of our approach.

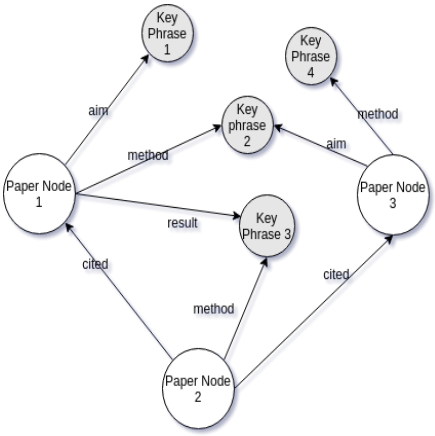

Figure 1: Structure of proposed Representation

Our research background, hypothesis, and motivation were presented in this section. In the following section, we describe proposed approach in detail. Finally, we present our datasets, experiments, and results and briefly summarize state-of-the-art approaches before concluding the paper.

2 Approach

2.1 Concept Extraction

Problem Definition: Given a target document d, the objective of the concept extraction task is to extract a list of words or phrases which best represent the aim, method and result of document d.
Prior work has solved the problem of extracting keyphrases and relations between them as a sequence labelling task. However, due to the non-availability of large annotated data for this purpose

limits this approach. Also this approach does not take advantage of the fact that more than 96 percent of phrases that form aim, method and result are noun phrases (Augenstein et al., 2017). Since we already have a defined set of candidates for the key phrases, we attempt this problem as multi-class classification problem. Given a document, we classify its phrases as *Aim*, *Method*, *Result*. Our approach is built on the observation that the semantics of the sentence of document d containing a phrase belonging to any of the concept type is similar across research papers. To capture this semantic similarity, we use k nearest neighbour classifier on top of state-of-the-art (Devlin et al., 2018) domain based word embeddings. We start by extracting features from a small set of annotated examples and used bootstrapping (Gupta and Manning, 2014) for extracting new features from unlabeled dataset. Figure 2 shows our pipeline.

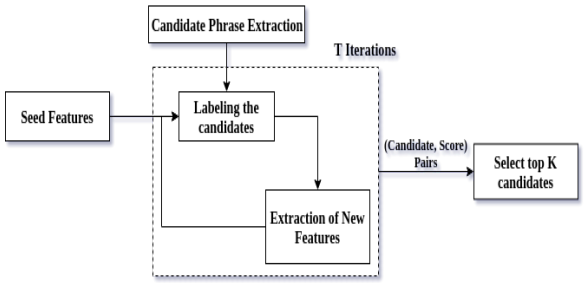

Figure 2: Proposed Method

Following are some of the terminologies which will be used throughout the paper that follows:

- *Candidate phrases*: Phrases present in the target document d which will be considered for labeling.
- *Concept mention*: Phrases labeled as either aim, method or Result in the labeled dataset.
- *Parent sentence of a phrase p*: The original sentence in target document to which the candidate phrase/concept mention p belongs to.
- *Left context phrase(S,p)*: The part of the parent sentence S before the occurrence of the candidate phrase p or concept mention.
- *Right context phrase(S,p)*: The part of the parent sentence S after the occurrence of the candidate phrase p or concept mention.
- *Left Context Vectors(p)*: Vector representations of left context phrase p.
- *Right Context Vectors(p)*: Vector representations of right context phrase p.
- *Feature Vectors*: Tuple of Left and Right Context Vectors which is being used as features to label candidate phrases.
- *Feature Score*: Each feature vector has an associated feature score between 0 and 1 that represents the confidence of it being a representative of the class. Seed features have a feature score of 1.
- *Support Score of candidate phrase p for class c*: Every phrase is assigned a support score for all classes that represents the confidence that the phrase belongs to that class.

Seed Feature Extraction: In this step, we extract features for each of the concept type using the small set of annotated examples. For each concept mention in the annotated examples, we construct left context vector *lcv* and right context vector *rcv*. These lcv and rcv then form part of the features for the class to which concept mention belongs to. Phrase embeddings are generated using pre-trained BERT model (Devlin et al., 2018) fine-tuned on DBLP research papers dataset. Details of BERT training and datasets used for seed feature extraction are given in the Experiments Section.

Candidate Phrase Extraction: To limit the search space of phrases, we propose to use noun phrases present in the Title and Abstract of document d as candidate phrases. For citation contexts, named entities form a better set of candidates as shown by (Ganguly and Pudi, 2016). However different named entities can be linked to different papers cited in the same citation context. So it becomes essential to first identify which entity *e* corresponds to which cited paper *cp* and then use the proposed algorithm to classify *e* as aim/method/result for the corresponding paper *cp*. For the above purpose, we use entity-citation linking algorithm (Ganguly and Pudi, 2016). The matching function iterates over entities and citations to get their closeness score. After the scoring step, a two-step pruning is performed. It first takes all the citations and keeps a list of the closest entity per citation. Then it takes the remaining entities and keeps only the closest citations per entity. Finally, we get a list of tuples where each element contains a unique entity matched with its citation. Only the entities which are present in this list of tuples are considered as candidate phrases.

Labeling Candidate Phrases: For labeling candidates in iteration *i*, we use k-NN. The algorithm for labeling candidate phrases is presented in Algorithm 1.

Algorithm 1: Label Candidate Phrases

1. For each sentence s in document d in the dataset, $p \leftarrow$ unlabeled Phrase in sentence s.

2. Let *lcv* be the left context vector and *rcv* be the right context vector corresponding to phrase p in sentence s.

3. Find nearest neighbours of lcv and rcv from the feature vectors that are atmax distance *r*. Let the nearest neigbours corresponding to lcv be *lnn or left nearest neighbours* and rcv be *rnn or right nearest neighbours.*

4. If the size of both lnn and rnn is less than the minimum number of neighbours required for classification *k* then the phrase can not be labeled in this iteration and we move to the next phrase.

5. Else we take k nearest neighbours for both the lcv and rcv and the support score of the phrase for class *c* is calculated as follows :

$N = \{n | n \in$ Top k Neighbours of lcv or rcv and $label(n) = c\ \}$

$$supportScore(p, c) = \sum_{n \in N} featureScore(n)$$

6. Then the predicted class for phrase p is $\arg\max_c supportScore(p, c)$.

Finally after T iterations, unlabeled candidate phrases are discarded.

Extraction of new features: For each phrase p assigned class c in any of the iterations, we generate context vectors *lcv* and *rcv*. We define the feature score corresponding to the context vectors of phrase p labeled as class c as:

$$featureScore(p) = \frac{supprtScore(p, c)}{\sum_{c'} supportScore(p, c^{'})}$$

For each class, the context vectors are sorted based on their feature score and top 5000 are taken as feature vectors.

Final Selection: For each document, we take top t phrases (based on their *supportScore*) for each class as the final output of our system.

2.2 Graph Construction

Graph definition: We build a graphical representation by using the extracted concepts and citation graph. Our graph has the following types of nodes and edges:

Paper nodes: These are the original paper nodes in the citation graph. Each paper node has metadata related to the paper like dblp id, title, authors, conference, year of publication.

Entity nodes: These nodes are the phrases extracted in the concept extraction step.

Cited_by relation: A cited_by relation is defined between paper nodes p_i and p_j if paper p_i has cited p_j.

Aim relation: Aim relation is defined between a paper node p_i and entity node e_i if e_i was extracted as aim concept for p_i.

Method relation: A method relation is defined between a paper node p_i and entity node e_i if e_i was extracted as method concept for p_i.

Result relation: A result relation is defined between a paper node p_i and entity node e_i if e_i was extracted as a result concept for p_i.

Construction of Graph: A major challenge in the construction of graph using phrases extracted in concept extraction step is merging of phrases with the same meaning. For the purpose of entity node merging, we do the following:

1. We group the papers according to the conference in which they were published. Then $\forall$ papers in the same group, we cluster their extracted phrases by running DBSCAN (Ester et al., 1996) over vector space representations of these phrases. The clusters are created based on lexical similarity which is captured by cosine distance between phrase embeddings. The intuition behind clustering phrases conference wise is that the research papers in a conference have same domain and thus phrases with high lexical similarity belonging to a particular conference are much more likely to mean the same as compared to phrases across conferences. This helps to avoid error as in the example : 'real time intrusion detection' in security domain and 'real time object detection' in computer vision domain are very different from each other but they may be clustered together by DBSCAN algorithm based on lexical similarity if DBSCAN is run on all the papers in the dataset at once.

2. Clusters merging across conferences: A cluster i belonging to conference c_1 and a cluster j belonging to conference c_2 are merged if they have any common phrase. This is done to capture the fact that there can be more than one conference

on same domain and hence some of their clusters should be merged if they correspond to same term or phrase. For example, both NAACL and ACL have papers on machine translation and therefore the individual clusters of these conferences corresponding to machine translation should be merged.

Finally we get clusters such that phrases in each cluster have the same meaning. We add only one entity node to the graph for each cluster. We define the relation type between a paper node and an entity node based on the label of the entity (phrase inside the entity node) for the corresponding paper as identified in Concept Extraction step.

3 Experimental Setup

Dataset Creation: All the experiments were conducted on DBLP Citation Network (version 7) dataset. This dataset is an extensive collection of computer science papers. DBLP only provides citation-link information, abstract, and paper titles. For the full text of these papers, we use the same dataset as have been used by (Ganguly and Pudi, 2017). This dataset is partly noisy with some duplicate paper information, and there is a lack of unique one-to-one mapping from the DBLP paper ids to the actual text of that paper. During the creation of our final dataset, we either pruned out ambiguous papers or manually resolved the conflicts. We came up with a final set of 465,355 papers from the DBLP corpus for which we have full text available. Since we need papers that are connected via citation relations, we prune our dataset by taking only the largest connected component in the citation network while considering the links to be bidirectional. We get 332,793 papers having 1,508,560 citation links. For extraction of citation context, we used Parscit (Prasad et al., 2018). For the papers for which abstract was not available in the DBLP dataset, we use the one extracted by Parscit.

Phrase embeddings: For vector representation of a phrase, we use BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding as proposed in (Devlin et al., 2018). We use the pre-trained model BERT-Base, Uncased: 12-layer, 768-hidden, 12-heads, 110M parameters available publically. We fine tune the model on our DBLP research paper dataset. Complete text of papers after cleaning has been used for the purpose of fine tuning. The model is fined tuned on total of 20970300 sentences with max sequence length as 128 and learning rate as 2×10^{-5}. For generating the phrase embedding we use second last layer as the pooling layer with pooling strategy as reduced mean.

Concept Extraction: (a) For the purpose of seed feature generation we use the following two publicly available datasets :

(i) SemEval 2017 Task 10 dataset (Augenstein et al., 2017): It contains 500 scientific paragraphs from physics, material science and computer science domain, each marked with keyphrases and each keyphrase is labelled as TASK, PROCESS and MATERIAL. The concepts of TASK and PROCESS in this dataset closely relates to our definition of AIM and METHOD. This complete dataset is used for seed feature extraction.

(ii) Gupta and Manning(2011) introduced a dataset of titles and abstracts of 474 research publications from ACL Anthology annotated with phrases corresponding to FOCUS, TECHNIQUE and DOMAIN. Their definitions of FOCUS and TECHNIQUE closely relate to our definitions of AIM and METHOD respectively. We divided this data into two parts- one is used as training data for seed features extraction having 277 papers and another as test data for evaluation purposes having 197 papers.

These two datasets helped to build seed features for AIM and METHOD category. We removed the papers from SemEval dataset which overlapped with (Gupta and Manning, 2011).

For RESULT, we manually annotated titles and abstracts of 100 research publications in computer science domain.

(b) While generating vector encoding for context phrases, we limit the length of the context phrase to 25 in-order to handle very long sentences. We used cosine distance to measure distance between vector representation of the phrases.

(c) It may be possible that there are more than one concept mention in a sentence. To nullify the effect of other concept mentions, we generated the seed features list in two ways:

- Take the left context phrase and right context phrase and generate their vector representation. This is called as *unmasked feature list*.
- We mask the other candidate phrases C in the left and right context phrase of candidate c_i

k	r	t	f1 score	precision	recall
30	0.65	3	**40.66**	46.04	**36.41**
60	0.65	3	40.47	**52.60**	32.88
40	0.65	3	40.38	48.65	34.51
40	0.60	4	40.06	47.12	34.84
30	0.75	4	38.38	41.95	35.37

Table 1: f1, precision & recall score for AIM concept

k	r	t	f1 score	precision	recall
40	0.85	20	**32.58**	22.65	58.1
30	0.75	17	30.81	21.12	56.89
30	0.90	14	30.87	**23.78**	44
30	0.80	25	31.16	20.72	**62.77**
30	0.65	15	30.69	21.35	54.6

Table 2: f1, precision & recall score for METHOD concept

before generating their embedding. This is called as *masked feature list*.

Experiments were done for masked and unmasked feature lists separately.

(d) As number of phrases added per iteration decreased substantially after iteration 5, we ran only 5 iterations of bootstrapping algorithm for all the experiments.

(e) We experimented with different values of distance r and k. We observed that in general precision increases with increase in value of k and recall increases with decrease in value of r.

Evaluation: For evaluating our results, we use the labeled dataset made available by (Gupta and Manning, 2011). We used 197 out of 474 papers for evaluation purpose. We calculate precision, recall and f1 score for each class. However, as *Result* phrases were not annotated in that dataset, we could evaluate only for *Aim* and *Method*. We compare our proposed approach with (Tsai et al., 2013) which ran the bootstrapping algorithm for a similar problem but used n-gram based features. They reported results for ACL Anthopology Network(AAN) Corpus (Radev et al., 2013). We ran their algorithm on our dataset with parameter tuning as mentioned by them.

4 Results and Discussion

4.1 Concept Extraction

We got the best results for parameter values, r = 0.65 and k = 60. Our bootstrapping algorithm gave output for 332,242 out of 332,793 papers. In Table 1, we report the top five scores for *Aim* for different parameters. Top ten scores for both aim and method concept were given by unmasked feature list. Therefore mask feature list results have not been shown. In Table 2 we report the top five scores for *Method* on different parameters. Table 3 and 4 compares our scores with that of (Gupta and Manning, 2011) and (Tsai et al., 2013). Table 5 compares our scores with the score computed for (Tsai et al., 2013) approach on our dataset.

Approach	f1 score	precision	recall
GM (2011)	30.5	46.7	36.9
(Tsai et al., 2013)	48.2	48.8	48.5
Our Approach	32.58	22.65	**58.1**

Table 3: Comparison with state-of-the-art for METHOD Concept

Approach	f1 score	precision	recall
(Tsai et al., 2013)	8.26	31.37	4.761
Our Approach	**40.66**	46.04	**36.41**

Table 4: Comparison with state-of-the-art for AIM Concept on DBLP dataset

Approach	f1 score	precision	recall
(Tsai et al., 2013)	18.0	50.70	10.94
Our Approach	**32.58**	22.65	58.1

Table 5: Comparison with state-of-the-art for Method Concept on DBLP dataset

Our proposed algorithm was able to extract phrases from scientific articles in a large dataset in semi-supervised manner with f1 score comparable to the state-of-the-art. Our f1 score was lower as compared to (Gupta and Manning, 2011) (Tsai et al., 2013). However, our recall was consistently higher. Our precision was perhaps low as we were considering only the noun phrases whereas such limitation was not there while annotating the test corpus. They (Gupta and Manning, 2011) (Tsai et al., 2013) used hand crafted features for AAN Corpus whereas our features were extracted algorithmically starting from a small annotated dataset containing multiple domains such as physics, material science and computer science. Table 5 shows the scalability of our approach. Tsai et al. (2013) bootstrapping algorithm could not give a decent score when ran on our multi-domain

dataset because phrases could not be extracted for most of the papers.

4.2 Graph Construction

Total number of unique phrases produced by the proposed algorithm are 565,031. Using DBSCAN we form 63,638 clusters having 266,015 phrases. Our final graph contains 332,242 paper nodes, 362654 entity nodes, 483899 aim relations, 982396 relations and 661 result relations. We store our graph in Neo4j database (Webber and Robinson, 2018). A small sample from our constructed graph is shown in figure 3. We can see that result relations are quite few as compared to method and aim relations. This is mainly because of less number of seed features for Result due to less annotated data as compared to Aim and Method.
The constructed graph can summarize the research community in the following way:
(i) All the papers on a particular topic can be accessed by just finding the entity node corresponding to the topic in the graph. The associated papers can also be differentiated on the basis of whether the topic appears as aim or method or result in the paper. This can also help in academic search and recommendation.
(ii) A field can be summarized by finding all the *methods* used in the field and applications of field by finding all the *aims* where the field has been used as *method*.
(iii) Trend Analysis, conference proceedings summarization, or summarization of a particular author's work can be done using the meta data in the paper node.
Neo4j provides interface for all kind of queries required for the above applications. The queries are out of scope of this paper.

5 Trend Analysis

We studied the field of computational linguistics and computer vision.
Computational Linguistics: We studied the growth and decline of following topics on the basis of relative number of papers published on each topic over a period of years: *summarization*, *word sense disambiguation* and *machine translation*. Papers are included from NAACL and ACL conferences from 1990 to 2012. Figure 4 and 6 show an example of trends as extracted from our constructed knowledge graph. Figure 6 shows transition of a topic from aim to method concept in the domain.

Computer Vision: We studied the growth and decline of following topics on the basis of relative number of papers published each topic over a period of years: *human pose detection, image segmentation* and *3d reconstruction*. Papers are included from CVPR, ECCV, ICCV and ICPR conferences from 1990 to 2012. Figure 5 and 7 show an example of trends as extracted from our constructed knowledge graph. Figure 7 shows transition of a topic from aim to method concept in the domain.

Meaningful results in the analysis for both the communities show the scalability and domain independence of our approach.

6 Related Work

There has been growing interest in studying automatic methods of information extraction from scientific articles. Our work maps to mainly two types of problems - Extracting keyphrases, concepts, and relations between them and extracting structured information in the form of knowledge graph from scientific literature.
Keyphrase extraction specifically from scientific articles started with SemEval 2010 Task 5 (Kim et al., 2010) which was focused on automatic keyphrase extraction from scientific articles and prepared a dataset of 284 articles marked with keyphrases. Gollapalli and Caragea (2014) studied the keyphrase extraction problem in an unsupervised setting. They extracted candidates from the title, abstracts and citation contexts and used Page Rank (PAGE, 1998) to give a score to the candidates. Gupta and Manning (2011) first proposed a task that defines scientific terms for 474 abstracts from the ACL anthology (Radev et al., 2013) into three aspects: domain, technique, and focus. They applied template-based bootstrapping on title and abstract of articles to tackle the problem. They used handcrafted dependency based features. Based on this study, (Tsai et al., 2013) improved the performance by introducing hand-designed features to the bootstrapping framework. Our system beats their systems in terms of recall for both aim and method concepts. Also, we worked on larger multi-domain dataset. SemEval 2017 Task 10 (Augenstein et al., 2017) focused on mention level keyphrase identification and their classification into three categories - TASK, PRO-

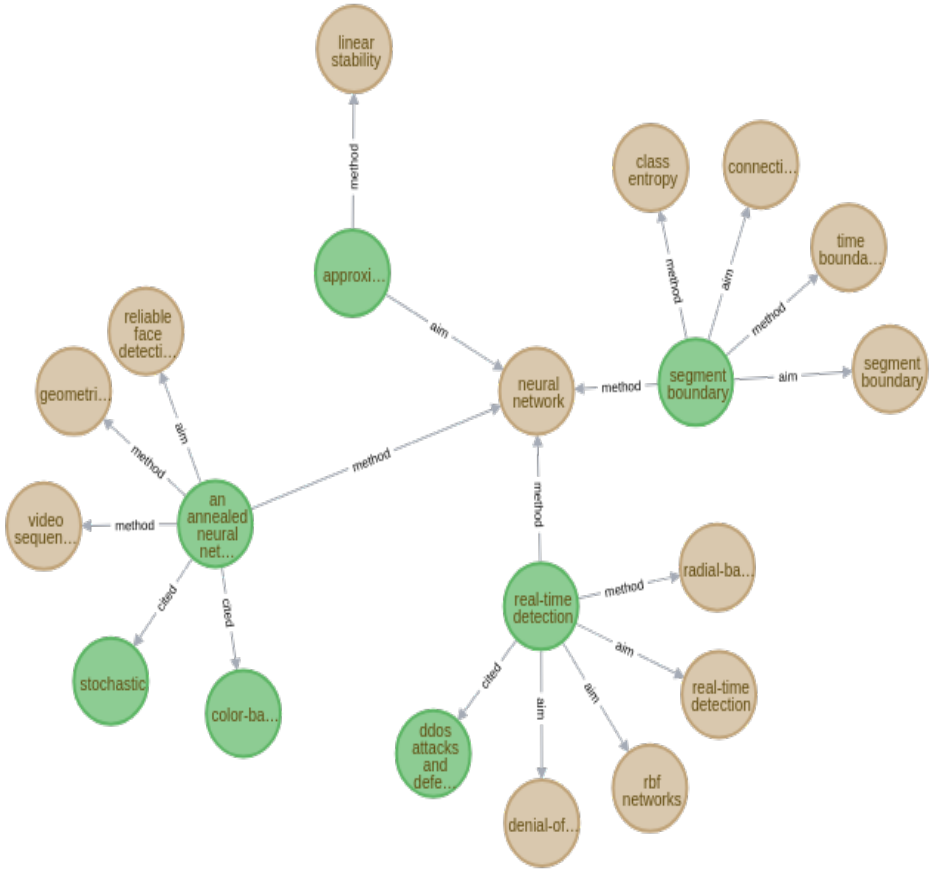

Figure 3: Sample from our constructed graph. Green nodes correspond to research papers and brown nodes correspond to extracted phrase entities.

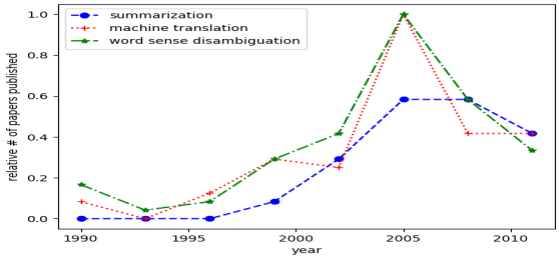

Figure 4: Growth and decline of research in different topics in computational linguistics

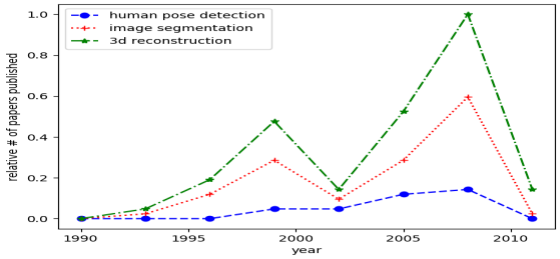

Figure 5: Growth and decline of research in different topics in Computer Vision

CESS, and MATERIAL. They prepared an annotated dataset comprising of 500 papers from Material Science and Computer Science journals. Many systems (Ammar et al., 2017) (Tsujimura et al., 2017) solved the problem in a supervised manner. Top system (Ammar et al., 2017) modeled the problem as a sequence labeling problem. (Tsujimura et al., 2017) trained LSTM-ER on that dataset. However, these supervised systems require a large amount of training data, in the absence of which they tend to overfit. Our semi-supervised method can work using a small set of annotated documents for initial features.

There is also an ongoing work on constructing knowledge graph from the scientific literature. Sinha et al. (2015) builds a heterogeneous graph consisting of six types of entities: field of study, author, institution (the affiliation of the author), paper, venue (journal and conference series) and event. Ammar et al. (2018) focussed on constructing literature graph consisting of papers, authors, entities nodes and various interactions between

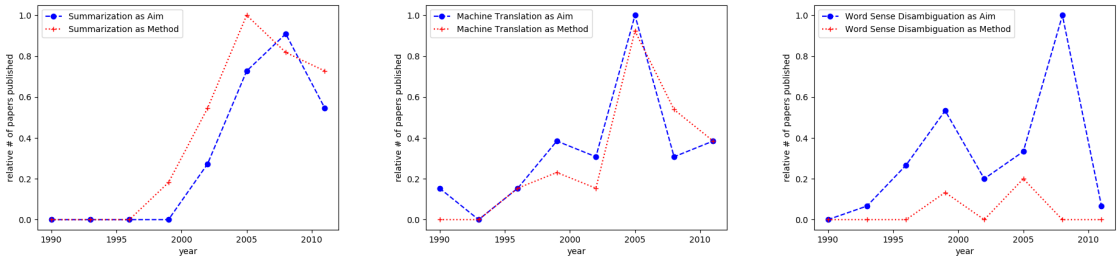

Figure 6: Transition from aim to method for 1. Summarization 2. Machine Translation 3. Word Sense Disambiguation

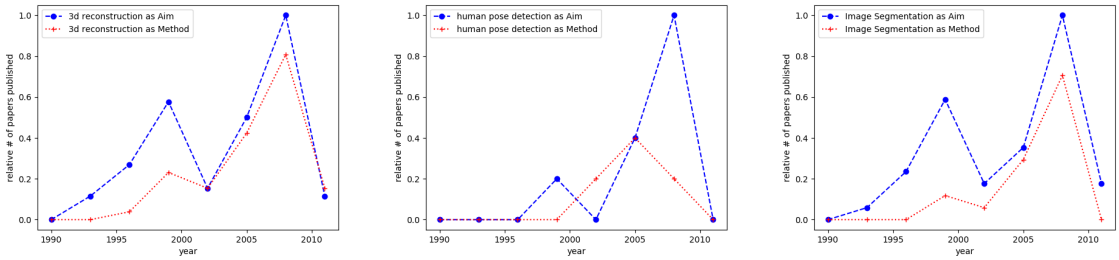

Figure 7: Transition from aim to method for 1. 3d reconstruction 2. Human pose-detection 3. Image Segmentation

them (e.g., authorship, citations, entity mentions). Luan et al. (2018) developed a unified framework for identifying entities, relations, and coreference clusters in scientific articles with shared span representations. They used supervised methods by creating a dataset which included annotations for scientific entities, their relations, and coreference clusters for 500 scientific abstracts from AI conferences proceedings. Our knowledge graph is more straightforward to build. Also, it is built upon the citation graph due to which it retains the vital citation information which is an integral part of the research community.

Conclusion

This work propose semi-supervised techniques for identifying *Aim, Method* and *Result* concepts from scientific articles. We show how these concepts can be introduced in the citation graph to graphically summarize the research community and the various applications of the graphical representation thus formed. We show the domain-independence of our approach as :- a) Seed features from one domain (physics, material science from SemEval dataset) were used to extract concepts for another domain (computer science papers from DBLP dataset), b) Meaningful results for two distinct communities as section 5. We also experimentally show the scalability of our approach and compared the results with the state-of-the-art.

References

Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, Rodney Kinney, Sebastian Kohlmeier, Kyle Lo, Tyler Murray, Hsu-Han Ooi, Matthew E. Peters, Joanna Power, Sam Skjonsberg, Lucy Lu Wang, Chris Wilhelm, Zheng Yuan, Madeleine van Zuylen, and Oren Etzioni. 2018. Construction of the literature graph in semantic scholar. *CoRR*, abs/1805.02262.

Waleed Ammar, Matthew Peters, Chandra Bhagavatula, and Russell Power. 2017. The ai2 system at semeval-2017 task 10 (scienceie): semi-supervised end-to-end entity and relation extraction. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 592–596, Vancouver, Canada. Association for Computational Linguistics.

Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. 2017. Semeval 2017 task 10: Scienceie - extracting keyphrases and relations from scientific publications. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 546–555. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and

Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press.

Soumyajit Ganguly and Vikram Pudi. 2016. Competing algorithm detection from research papers. In *Proceedings of the 3rd IKDD Conference on Data Science, 2016*, CODS '16, pages 23:1–23:2, New York, NY, USA. ACM.

Soumyajit Ganguly and Vikram Pudi. 2017. Paper2vec: Combining graph and text information for scientific paper representation. In *Advances in Information Retrieval*, pages 383–395, Cham. Springer International Publishing.

Sujatha Das Gollapalli and Cornelia Caragea. 2014. Extracting keyphrases from research papers using citation networks. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, pages 1629–1635. AAAI Press.

Sonal Gupta and Christopher Manning. 2011. Analyzing the dynamics of research by extracting key aspects of scientific papers. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1–9. Asian Federation of Natural Language Processing.

Sonal Gupta and Christopher D. Manning. 2014. Improved pattern learning for bootstrapped entity extraction. In *CoNLL*.

Kokil Jaidka, Muthu Kumar Chandrasekaran, Sajal Rustagi, and Min-Yen Kan. 2018. Insights from cl-scisumm 2016: the faceted scientific document summarization shared task. *International Journal on Digital Libraries*, 19(2):163–171.

Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. 2010. Semeval-2010 task 5: Automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 21–26, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. *CoRR*, abs/1808.09602.

L. PAGE. 1998. The pagerank citation ranking : Bringing order to the web. *http://www-db.stanford.edu/backrub/pageranksub.ps*.

Animesh Prasad, Manpreet Kaur, and Min-Yen Kan. 2018. Neural parscit: a deep learning-based reference string parser. *International Journal on Digital Libraries*, 19(4):323–337.

DragomirR. Radev, Pradeep Muthukrishnan, Vahed Qazvinian, and Amjad Abu-Jbara. 2013. The acl anthology network corpus. *Language Resources and Evaluation*, pages 1–26.

Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June (Paul) Hsu, and Kuansan Wang. 2015. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15 Companion, pages 243–246, New York, NY, USA. ACM.

Chen-Tse Tsai, Gourab Kundu, and Dan Roth. 2013. Concept-based analysis of scientific literature. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, CIKM '13, pages 1733–1738, New York, NY, USA. ACM.

Tomoki Tsujimura, Makoto Miwa, and Yutaka Sasaki. 2017. Tti-coin at semeval-2017 task 10: Investigating embeddings for end-to-end relation extraction from scientific papers. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 985–989, Vancouver, Canada. Association for Computational Linguistics.

Jim Webber and Ian Robinson. 2018. *A Programmatic Introduction to Neo4J*, 1st edition. Addison-Wesley Professional.