18th International Conference on Information Fusion
Washington, DC - July 6-9, 2015

Discover Trending Domains using Fusion of Supervised Machine Learning with Natural Language Processing

Shilpa Lakhanpal[§] Ajay Gupta[§] Rajeev Agrawal[†]

[§]Western Michigan University, Kalamazoo, MI, U.S.A, [shilpa.lakhanpal, ajay.gupta]@wmich.edu
[†]North Carolina A&T State University, Greensboro, NC, U.S.A, ragrawal@ncat.edu

Abstract - In this paper, a new technique is presented for mining key domain areas from scientific publications. A domain refers to a particular branch of scientific knowledge and hence largely defines the theme of any scientific research paper. The proposed technique stems from a fusion of knowledge derived from natural language processing and machine learning. Some words or phrases are extracted based on their meaning inferred by the application of preposition disambiguation. These key words or phrases are then classified as domain areas using supervised learning. Various experiments and their analyses yield concrete results validating the efficacy and application of our methodology. The fusion technique therefore extracts an interesting aspect of research from scientific text and hence propounds a hybrid methodology for deriving meaning from underlying text. This approach thus takes a definitive step in advancing text analytics.

Keywords: Preposition Disambiguation, NLP, Supervised Classification, Naïve Bayes classifier.

1 Introduction

Text mining is a burgeoning field, which involves automating the extraction of knowledge from natural language. Natural language processing (NLP) is an indispensable assortment of techniques in this text mining process as it aims to automate the understanding of text by computers, in a way analogous to humans. NLP is hard as analysts encounter ambiguity at pragmatic, syntactic, semantic, phonetic and morphological levels. We describe a method to alleviate this ambiguity by combining techniques from supervised machine learning. This fusion approach merges techniques from two major scientific domains and yields impressive results.

Preposition sense disambiguation is an NLP technique where scientists [1, 2] have extracted major "senses" that are conveyed by prepositions in text. Using these senses or meaning of prepositions, we extract certain interesting phrases from text. These interesting phrases are subsequently classified using a Naïve Bayes classifier.

We apply our technique to textual data from scientific research papers published across various technical conferences which document the research endeavors of various scientists from around the world. The key interesting parts of each research paper that we extract are the domain areas of that paper.

A domain area of a scientific paper is the main topic or theme addressed in it. A domain area is a branch of a scientific field. For example, the field of Computer Science has domains such as networking, parallel computing, theory of computing, data mining, etc. Each domain can in turn have several subdomains, where the latter are considered domains as well, albeit smaller ones, and so on, creating a hierarchy of sorts. A scientific paper documents research endeavors for solving a specific problem within its domain area. Thus a problem-area is the current focus of the research of that paper. Related problem-areas are encompassed together into a larger set called domain. Taking an example of the hierarchical structure alluded to earlier, the root for the domain called Graph Theory will have all the problem-areas associated with graphs, and a subdomain called Routing will have problem-areas on routing problems such as Minimum Spanning trees, Travelling salesman problems, Hamiltonian circuits, etc.

These problem-areas are tackled by employing techniques or methodologies which are described in each paper. Note that the difference between a domain and a problem-area is not always well-defined. A problem-area that was initially a focus of a small amount of research, over time, gains traction. Researchers begin to explore it in detail and start focusing on newer sub-problem-areas. A problem-area may over time thus become a domain in its own right. Hence, for the scope of this paper domain and problem-area are synonymous, as our goal is to segregate techniques from domains (or problem-areas).

We have pitched the initial idea of this technique in our preliminary work [3]. In this paper we extend our ideas and present our exhaustive technique and validate it by presenting various experimental results along with detailed analyses and future scope. The rest of the paper is organized as follows. Section 2 briefly discusses related work. Section 3 provides an explanation and definitive argument toward the uniqueness and utility of our technique. Section 4 contains our extensive results and analyses. Finally Section 5 offers conclusions and future work that we target beyond this current work.

2 Related Work

Analyzing the focus of research by extracting information from research database is becoming an active field. Techniques from NLP domain have been employed toward this goal. A bootstrapping learning technique has been proposed in [4] to extract items such as domain areas, focus of research and techniques from research papers. Using dependency trees and starting with some handwritten semantic patterns in three categories of domains, focus, and techniques, their methodology learns new patterns. Although the work provides key insights, their results are not that encouraging as they themselves claim that their system failed to correctly address patterns which it found to be outside their three pre-defined categories [4]. Analysis of their results indicates that their technique for domain extraction has high recall but suffers from low precision [4]. This indicates that although they are able to retrieve domains, they also incorrectly mark non-domains as domain areas. Our approach does not explicitly use NLP per se but fuses NLP and supervised learning to obtain good results of high precision and high recall for labelling domains.

Supervised learning for text classification has been widely used in applications of NLP. Hidden Markov Models (HMMs) are statistical tools for modeling generative sequences that can be characterized by an underlying process generating an observable sequence [5]. In NLP, they are used to mark the part-of-speech category of various words in text. The HMM model is a stochastic analog of finite state automaton, with probabilistic transitions between states. HMMs have been used for sentence classification [6], where the preferred sequential ordering of sentences in the abstracts of "Randomized Clinical Trial" papers, facilitated its use. The sentences in the abstract are supposed to be ordered in sequence of "background," "objective," "method," "result" and "conclusion" [6] and model-states are aligned to these sentence types. Our approach does not depend on a generative process as the "domain", "problem-area" and technique can occur in any random order in a title. Hence our approach targets more generic solutions.

In our previous work [7], we extracted the prevalent trends of research using a phrase-based approach. We created a simple but intuitive technique to analyze the titles of a collection of research papers. A title was first mined to extract its constituent phrases, which were enclosed between or delimited by well-defined stopwords. By counting the frequency of phrases across the collection of research papers, it was possible to generate the most frequently occurring phrases, and hence the most frequent trend in prevalent research. The titles tend to be unique, and hence the ordered sequential left to right structure of phrases may be restrictive as we did not account for the permutations. In this paper, we take our work much further by incorporating a fusion of NLP with intelligent machine learning techniques to extract meaningful domain areas from research papers.

3 Our Approach

Our technique extracts theme from each research paper in the form of its domain. We derive interesting phrases based on their placement in the vicinity of certain prepositions by using results of preposition disambiguation. Even though a research paper has structure in terms of its division into sections such as abstract, introductions, etc., still the text in these sections is just a bag of words for a computer. Hence we train a computer algorithm to classify the interesting phrases as to whether they are domains or not. Therefore phrases are accorded a meaning and this meaning is derived exactly as the respective authors themselves wished to convey. Besides the quality of fusing knowledge from NLP and supervised learning, our technique effectively derives meaning of text without explicitly using the constructs of NLP.

3.1 Definitions

We define some important concepts as they shall be used for discussion. These standard definitions are reproduced from [3], [8] to enhance the readability of this paper.

Word: A single and distinct element of language which has a meaning and is used with other words to form a sentence, clause or phrase

Stopword: Word in the language, such as "and", "the", which is very common, but is not very useful when selecting text that answers a user's query

Sentence: A sequence of words that is complete in itself, containing a subject and predicate, conveying a statement, question, exclamation, or command, and consisting of a main clause and, optionally, one or more subordinate clauses

Clause: A unit of grammatical organization next below the sentence in rank and in traditional grammar said to consist of a subject and predicate

Phrase: A small group of words standing together as a conceptual unit, typically forming a component of a clause

m-gram: A contiguous sequence of m words in a given sequence of words

Preposition: A word governing, and usually preceding, a noun or pronoun and expressing a relation to another word or element in the clause

Preposition with Intention Sense: The preposition that indicates that the phrase following it specifies the purpose (i.e., a result that is desired, intention or reason for existence) of an event or action

Phrase of Interest (Interesting Phrase): A phrase that follows a preposition with intention sense and ends before the next preposition in the clause or ends with the end of the clause

Derivative: Keyword or keyword phrase which has one or more words in common with an interesting phrase

Domain Word: A word that denotes or has a potential for naming a well-accepted domain area, or is a part of a phrase denoting a well-accepted domain area

3.2 Preposition Sense Disambiguation

A preposition as defined above expresses a relation between two elements of a clause. One relation can be conveyed by different prepositions depending on the context in which they are used. Conversely one preposition can convey different meanings. The position of prepositions in text and their contextual use can provide extremely useful insight into the meaning of text. Much work has been dedicated to extricate the "sense" or the "relation" conveyed by the presence of various prepositions within different group of words [1, 2]. We would like to explain the meaning of intention. For example the intention sense is communicated by the preposition "*for*" in the phrase "system *for* extracting data". According to the work in [1], the "complement" of the preposition conveys the "intention" or "purpose". In the English language the complement generally refers to a noun phrase, pronoun, a verb, or adverb phrase [8]. Another term used to denote the "complement" is called the "object" of the preposition as used in [2], who have identified an inventory which presents 32 different meanings, built on the "relations" established by the usage of prepositions in various settings. It may be noted that the 7 different senses [1] seem to encompass the 32 relations elicited by authors in [2]. Hence we chose to work with the senses of the prepositions. Authors of a technical paper may want to communicate the crux of their paper through their titles [9] most likely by using technical terminology while paying less attention to nuances of English language such as adverbs or pronouns [8]. Hence, for simplicity we pick the complement that will be delimited at the other end by the next preposition or end of the clause and define it as an "interesting phrase".

We have compiled a complete list of prepositions after reviewing several English handbooks. Careful study of the preposition senses narrowed down in [1] has allowed us to create our set of prepositions with intention sense, $\boldsymbol{PI}$, as depicted in equation (1). We denote each preposition in this set as $\boldsymbol{p_i}$. We denote all other prepositions as $\boldsymbol{p_o}$.

$$\boldsymbol{PI} = [\textit{for}", \textit{to}", \textit{towards}", "\textit{toward}"] \quad (1)$$

The complement, $\boldsymbol{C}$, is a phrase that is extracted based on the permutations of $\boldsymbol{p_i}$ and $\boldsymbol{p_o}$ in a clause. $\boldsymbol{E}$ denotes the end of the clause. Equation (2) depicts the relevant permutations and the corresponding complement. It should be assumed that there is a space between each two consecutive words, even though these spaces are not explicitly represented in the equations.

$$\begin{matrix} \boldsymbol{p_i C p_o} \\ \boldsymbol{p_i C p_i} \\ \boldsymbol{p_i C E} \end{matrix} \quad (2)$$

3.3 Fusion of Title and Keywords

We start with the title of a research paper as the authors would probably want to highlight the goal of their research in their title [9, 10, 11, 12]. In order to relay their goal in as succinct form as possible yet making it comprehensive enough, they might include the underlying theme or main topic or the domain of their research. Since interesting phrases by their very definitions reflect the "purpose" or the "goal" in their respective sentences, we extract the interesting phrases from the titles. These interesting phrases in most cases shall hint upon the domains of the papers. Writing is largely subjective, and each author's perspective of the goal of his research dictates its representation. But in order to garner a wider audience, he might hint upon the larger domain.

In the keyword section of a research paper, the authors list the key phrases or key words of their documents [13]. Since titles tend to be unique, their constituents may not by themselves be good representatives of general domain areas. The keywords on the other hand are more commonly and widely used, well accepted set of general terms that various authors use to label their work. Hence they serve as generic terms which authors might use to depict their domains, problem-areas and techniques. We combine the knowledge gained from the interesting phrases from the title with the keywords and key phrases of the respective paper. Thus essentially we are using the important sections of a paper to get at the major theme of that paper. To retrieve the generic aspect of the interesting phrase, we retain those keywords and/or key phrases that have any words in common with the interesting phrase.

3.3.1 Extracting Derivatives

Grammatically, the title of a paper could be a sentence, clause or phrase. We scan each title, $\boldsymbol{T_{ti}}$ to find the prepositions with intention sense.

In equation (3), we have listed various example permutations of $\boldsymbol{p_i}$ and $\boldsymbol{p_o}$ within an example title, $\boldsymbol{T_{ti}}$. Note that in a research paper title, one or more instances of $\boldsymbol{p_i}$ and $\boldsymbol{p_o}$ can occur in several, all or more permutations than the ones listed in equation (3).

$$\boldsymbol{T_{ti}} = w_1..w_{j-2}\boldsymbol{p_i}w_j..w_{k-2}\boldsymbol{p_i}w_k..w_{l-2}\,\boldsymbol{p_o}w_l..w_{m-2}\,\boldsymbol{p_i}\,w_m..w_n \quad (3)$$

Next, we extract those interesting phrases that follow any instance of a $\boldsymbol{p_i}$ preposition and are delimited at the other end by any instance of a $\boldsymbol{p_i}$ or $\boldsymbol{p_o}$ or the end of the title. For title, $\boldsymbol{T_{ti}}$, the phrases of interest, $\boldsymbol{PHOI_{ti}}$ are listed in equation (4).

$$\boldsymbol{PHOI_{ti}} = \begin{matrix} w_j w_{j+1}..w_{k-3}w_{k-2} \\ w_k w_{k+1}..w_{l-3}w_{l-2} \\ w_m w_{m+1}..w_{n-1}w_n \end{matrix} \quad (4)$$

The next step involves finding an intersection between phrases in set, $\boldsymbol{PHOI_{ti}}$ with the keyword section, $\boldsymbol{KW_{ti}}$ of that particular paper. In this step, we retain those keyword or keyword phrases which have one or more words in

common with the interesting phrases. This resultant set, $\boldsymbol{D_{ti}}$ or the derivative becomes the main element of our analysis. Equation (5) lists an example $\boldsymbol{KW_{ti}}$ set of paper with title, $\boldsymbol{T_{ti}}$.

$$\boldsymbol{KW_{ti}} = \begin{matrix} w_j w_{k+1} \\ w_{p-3} w_{p-2} \\ w_q w_{q+1} w_{q+2} \\ w_m w_{m+1} w_{k-1} \end{matrix} \quad (5)$$

Note that words in the key phrases appearing in the keyword set could be in any order. We would like to stress that our approach considers a word by itself as a stand-alone entity and hence the order of words in the key phrases with respect to the interesting phrases does not matter. It is the word's appearance at strategic locations within the interesting phrase and the keyword section which clues us in to its importance in its part as the derivative. The interesting phrase already has a meaning based on its derivation and its words find accentuated generic meaning when they also occur within the keyword section. Hence our technique *infers* the meaning of a word without actually using a dictionary, thesaurus or even NLP.

Equation (6) lists the resultant derivative set $\boldsymbol{D_{ti}}$ of that paper.

$$\boldsymbol{D_{ti}} = \begin{matrix} w_j w_{k+1} \\ w_m w_{m+1} w_{k-1} \end{matrix} \quad (6)$$

3.4 Supervised Classification

Classification is the task of assigning one of a small number of discrete valued labels to the input data. We classify each derivative as a "Domain" or "Not Domain". Hence our classifier takes the approach of supervised learning as the training data (derivative) will be accompanied by labels indicating the class of the derivative.

We build a repository of subdomain areas in a major domain area of a scientific field through extensive research and analysis of important and trending topics across various scientific conferences and journals. These subdomains are considered domains as they are nodes in the hierarchical structure alluded to in Section 1. This repository consists of a list of single words or unigrams (1-grams). These unigrams either as stand-alone or as part of a phrase built from other members of this list represent well accepted domain areas. We may wish to point out that though such unigrams by themselves may sometimes not be domains, but them being a part of the topics from which they are derived, make them a domain word. We stress on the fact that this list contains well accepted domains as the latter have been obtained from credible sources viz. scientific conferences which are organized by experts in said scientific field.

We analyze each derivative, and if it has any word from this repository, we label the derivative as a "Domain". In case the derivative finds no match in the repository, that labels it as a "Not Domain". Thus, we analyze the list of derivatives and assign corresponding class labels to them. We reiterate that without knowing the actual meaning of a word, we are *inferring* its significance. Such as a word in the derivative is likely a domain word if is found in the repository of domains list.

The next step in creating the classifier is deciding what features of the derivatives are relevant.

3.4.1 Session Identifiers

A scientific conference has various sessions each of which assembles the papers dealing with similar topics in one group. Each such session is identified by a name which represents the topic of each group in a comprehensive yet succinct way. Hence logically this session identifier represents the domain of its group of papers. We process each derivative to see if it has any word in common with the session identifier. Any common word between the derivative and the session identifier sets the feature of the derivative as "Found in Session: True". No common word sets the feature as "Found in Session: False". An important point to be noted is that we do not restrict each derivative of a paper to the latter's respective session identifier. Rather we compare it across the entire set of session identifiers across the years of the conference under analysis and consider at least one match as a positive find and no match at all as negative. The reason we use the entire set is that grouping of the papers into each session and naming the session identifier is subjective and based on the conference committee's opinions and preferences.

3.4.2 Abstract Count

An abstract of a paper is written so as to contain the main elements of the paper in a synoptic form [14]. This makes it a likely section to contain the underlying theme and hence the domain area of the paper. Therefore the likelihood of any word of the derivative to be a domain word could be supported by its appearance in its respective paper's abstract. Since the domains are generic and different papers could share a domain area, hence we match words from each derivative across all papers in the data set. Therefore we count the abstracts containing at least one word of the derivative. This frequency becomes a relevant feature, because different abstracts containing the words of the derivative validate the importance of a derivative. If a derivative contains more domain words, it adds to its validity of becoming a domain as a whole. For example, a derivative "pattern recognition" has a count of 50, if "pattern" occurs in 30, "pattern recognition" occurs in 5 and "recognition" occurs in 15 abstracts.

We discretize the count of the abstracts as integer values from 1 to 5, after dividing the count values into groups of 5.

3.4.3 Naïve Bayes classifier

The Bayes rule in probability theory is represented in equation (7) [15].

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} \quad (7)$$

Our feature extractor functions create a feature set containing relevant feature values for all given derivatives. Since the appearance of a word of the derivative in a session identifier and in an abstract is independent, we use a Naïve Bayes classifier as it works well with independent features. We denote a feature vector as $\boldsymbol{X}$ and the class label as $\boldsymbol{Y}$. Our feature vector is represented in equation (8).

$$\boldsymbol{X} = [X_1\ X_2\ ] \quad (8)$$
$$where,\ X_1 = Found\ in\ Session\ Identifier$$
$$X_2 = Abstract\ Count$$

The class label $\boldsymbol{Y}$ takes binary values as represented in equation (9).

$$\boldsymbol{Y} = \begin{matrix} Domain \\ Not\ Domain \end{matrix} \quad (9)$$

We would like to model $\boldsymbol{P(X|Y)}$, where $\boldsymbol{X}$ is a feature vector, and $\boldsymbol{Y}$ is its associated label. Our task is demonstrated in Figure 1. It may be pointed out that feature X_1 is a binary attribute, while feature X_2 is a 5-valued attribute.

Naïve Bayes classifier

Task: Predict whether or not a derivative is a domain

Training Data:

$\mathbf{X} = (\ \mathbf{X_1}\ \ \mathbf{X_2}\ )\ \ \mathbf{Y}$

n rows

Found in Session Identifier	Abstract Count	
True	1	Domain
False	1	Not Domain
False	3	Domain
True	2	Domain
True	5	Not Domain
:	:	:

Figure 1: Naïve Bayes Classifier

In order to accurately estimate $\boldsymbol{P(X|Y)}$, we need to consider the number of parameters we must estimate given our $\boldsymbol{X}$ and $\boldsymbol{Y}$. Hence we need to estimate a set of parameters, $\boldsymbol{\theta_{ij}}$ given in equation (10), where $\boldsymbol{i}$ takes on (2 + 5) possible values (one for each of the possible vector values of $\boldsymbol{X}$), and $\boldsymbol{j}$ takes on 2 possible values. To calculate the exact number of required parameters, note for any fixed $\boldsymbol{j}$, the sum over $\boldsymbol{i}$ of $\boldsymbol{\theta_{ij}}$must be one. Therefore, for any particular value $\boldsymbol{y_j}$, and the 7 possible values of $\boldsymbol{x_i}$, we need compute only (7-1)=6 independent parameters. Given the two possible values for Y, we must estimate a total of 2*6 = 12 such $\boldsymbol{\theta_{ij}}$ parameters.

$$\boldsymbol{\theta_{ij} \equiv P(X = x_i|Y = y_j)} \quad (10)$$

Since Naïve Bayes works with the simplified assumption of conditional independence among the attributes, $\boldsymbol{P(X|Y)}$ is calculated using equation (11).

$$\boldsymbol{P(X|Y) = P(X_1|Y)\ P(X_2|Y)} \quad (11)$$

The conditional independence assumption reduces the number of parameters to be estimated from 12 to 4. Although this reduction is not dramatic enough for our case, we may wish to point out that it will be considerable when we apply the Naïve Bayes classifier to extract more knowledge from research papers. An example of this knowledge is the set of techniques applied in research papers. The reason for this is that the relevant features for techniques may be more in number, and additionally may have multiple values.

3.5 Our Technique Exemplified

We describe our approach using an example. Figures 2(a) and 2(b) depict use case diagrams portraying the steps to arrive at the derivative. We use data of a paper from the ACM Special Interest Group on Data Communication (SIGCOMM) 2013 conference.

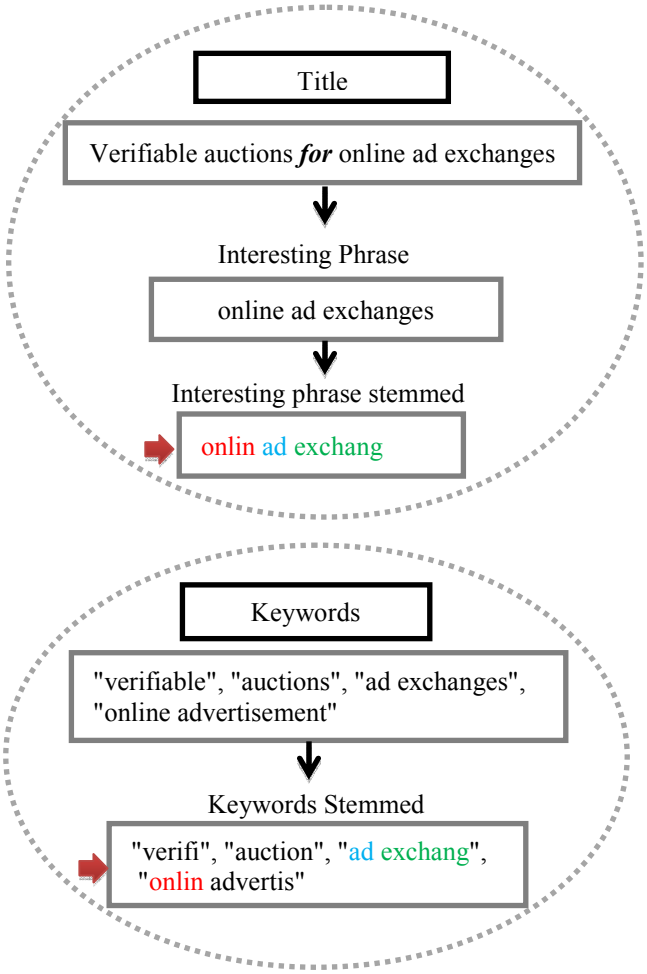

Figure 2(a): Use Case Diagram for our technique

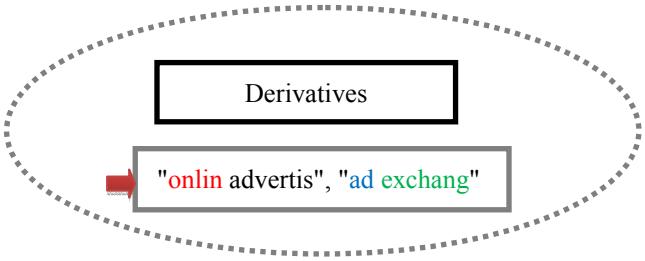

Figure 2(b): Use Case Diagram for our technique

Figure 3 depicts the processing for each derivative to find relevant features using all session identifiers and abstracts from all papers of SIGCOMM conference series from years 2010-2014.

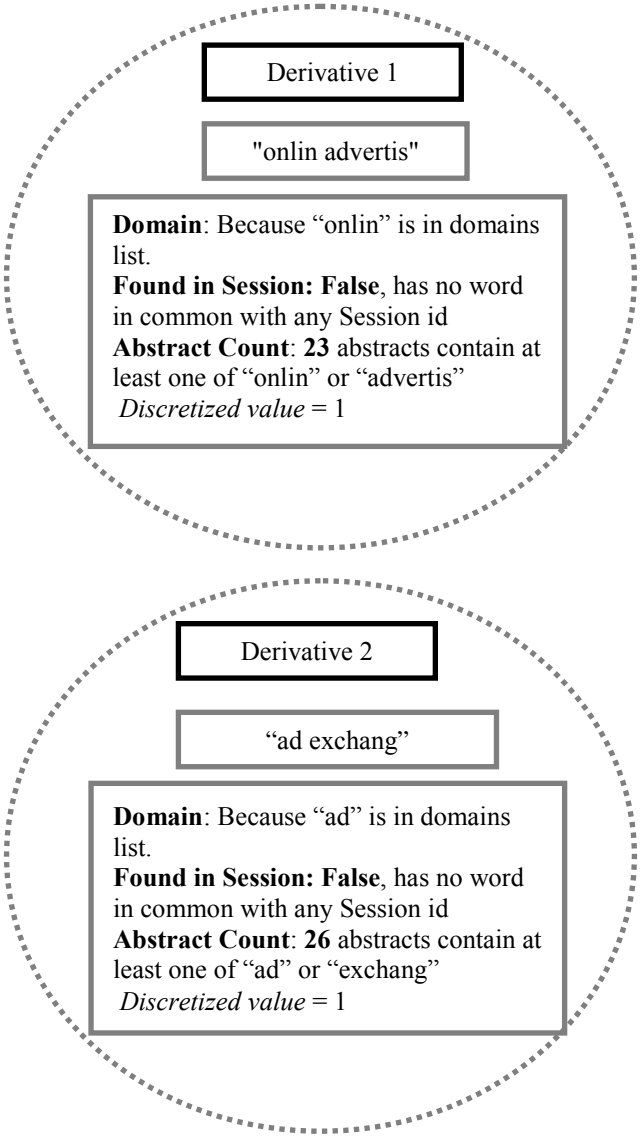

Figure 3: Processing each Derivative

Section 4 discusses the results of our experiments in detail.

4 Experimental Evaluation

We have programmed our technique using Python and some of its packages including NLTK. Although our approach is extendable to any scientific field, we test our technique on the research conferences in the field of Computer Science.

In order to create a repository of domain areas, our strategy is to collect the topics from the Calls for Papers (CFP) of top conferences of a large domain within Computer Science. CFPs for any conference contain topics under which papers are sought. Hence they are one of the definitive sources of domains, well-accepted by experts in the scientific field. These topics are in the form of sentences, clauses or phrases. We remove all the punctuations, stopwords and newline characters from these topics. This corpus is then stemmed, and each word hence becomes a domain word in our list of domains.

4.1 Datasets Used

In a set of experiments [3] on conferences on Data Mining, we collected the topics from the Calls for Papers sections from the IEEE International Conference on Data Mining series (ICDM), the IEEE International Conference on Data Engineering (ICDE), and the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD) from 2010-2014. For data analysis, we collected papers from ACM SIGKDD from years 2010-2014. This data includes 939 papers from all sessions including keynote, panel, demonstration, poster, industrial and government track apart from the regular research track sessions. We have extracted titles, session identifiers, and keyword lists for each of these papers. Of the 939 paper titles, 367 have prepositions with intention sense. Of the 367, we get 228 non-empty derivatives sets. These non-empty derivatives result when there is a match between the interesting phrase and the keyword list. From the 228 non-empty derivatives sets, we get 272 derivatives, because one derivatives set can have more than one derivative. It may be pointed out that meaningful session identifiers are a fairly new phenomenon and hence we only collect data from 2010-2014, where this phenomenon is more prevalent. In order to maintain parity, we have collected call for papers from ICDM, ICDE and SIGKDD only for the years 2010-2014.

The final dataset of 272 (ACM SIGKDD) derived as explained above is small at a first look, but the key thing to note here is that this is the derivate list. Once we have the derivatives which were derived using due process from their "respective" papers, the derivatives subsequently become the main element of analysis. We process the derivatives using the list of all session identifiers for reasons noted in Section 3.4.1. Moreover session identifiers have been rarely used in identifying true domains of papers; hence they

prove to be good sources of useful information. We also use the abstracts from all the papers of all the years of the conference under analysis. The reason simply is that authors exercise their choice in choosing titles and may or may not use prepositions with intention sense. But this no way implies that their domain is not the same as that of the authors that do use prepositions with intention sense. Hence we cannot restrict the "analysis" of our derivatives to only the abstracts of the papers from which they are derived. Also our point of contention was never the size of the dataset, rather the intelligence we derive from it, based on fusion of different sources of data.

Table 1 summarizes the count of the successive datasets as we progress in our analysis in various sets of experiments.

Conference	Titles	Titles with ***PI***	Derivatives
SIGKDD	939	367	272
SIGCOMM	414	136	99
ICDCS	369	139	113

Table 1: Count of successive datasets

After having extracted the feature sets for the derivative data as explained above, we divide them into a training set and a test set in the ratio of 70%-30% respectively. The training set is used to train a Naïve Bayes classifier.

To validate the efficacy of our technique we conducted a set of experiments on conferences on Computer Networks and Wireless Communication, where we created a domain list using topics from the Calls for Papers sections from the IEEE International Conference on Computer Communications (INFOCOM), the ACM International Conference on Mobile Computing and Networking (MobiCom), and the ACM Special Interest Group on Data Communication (SIGCOMM) from 2010-2014. We collected papers from ACM (SIGCOMM) from 2010-2014.

In a set of experiments on conferences on Distributed and Parallel Computing, we gathered a domain list using Call for Papers sections from IEEE International Conference on Distributed Computer Systems (ICDCS), the IEEE International Parallel and Distributed Processing Symposium (IPDPS) and the ACM Symposium on Principles of Distributed Computing (PODC) from 2010-2014. For data analysis, we collected papers from IEEE ICDCS from 2010-2014.

4.2 Results

The two most frequent and basic measures for information retrieval effectiveness are precision and recall. In binary classification, precision is the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. The precision and recall values are calculated using true positives, false positives, and false negatives which result from running the classifier on the test set. The formula is given in equation (12). True positives (TP), refer to the cases within the test set when domains are correctly identified, while false positives (FP) mean when certain "not domains" are labelled as domains. True negatives (TN) on the other hand correctly identify "not domains", while false negatives (FN) incorrectly label domains as "not domains".

$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{TP + FN} \quad (12)$$

The values for TP, FP, TN, and FN for one iteration of each dataset are listed in Table 2.

Conference	TP	FP	TN	FN	Precision	Recall
SIGKDD	52	2	21	6	0.963	0.8965
SIGCOMM	15	2	8	4	0.8823	0.7895
ICDCS	15	6	10	2	0.7143	0.8823

Table 2: TP, FP, TN, FN values for 1 iteration

Our technique has high precision and high recall as is demonstrated by average precision and recall values from the 100 iterations for each set of experiments. These values in percentages are tabulated in Table 3.

Conference	Precision	Recall
SIGKDD	95.54%	87.97%
SIGCOMM	90.42%	76.60%
ICDCS	77.15%	81.88%

Table 3: Average Precision and Recall

There is generally a tradeoff between precision and recall, where a higher value of one can be achieved at the cost of the other. Our technique scores as it generates fairly high values for both precision and recall.

The average accuracy of the classifier is tabulated in Table 4.

Conference	Accuracy
SIGKDD	87.05%
SIGCOMM	77.24%
ICDCS	74.33%

Table 4: Average Accuracy

5 Conclusion and Future Work

We have obtained very encouraging results from our technique. We have applied *fusion of NLP with supervised classification* and developed a methodology for extracting domains from scientific papers. We have used a *fusion of data* from different strategic sections of each paper. We have performed exhaustive experiments on different domains of computer Science and achieved good results validating our approach. Thus our approach opens exciting possibilities for developing a new genre of hybrid methodologies.

Our technique is tested in a controlled environment, where in we have restricted our experiments to the data from the field of Computer Science. We wish to extend it beyond Computer Science to other scientific fields such as the medical field, etc. Since we have used the structure of a research paper to our advantage, we wish to evaluate whether our idea fares well in the absence of such structure. We further wish to evaluate the scalability and the generalizability of our method. We have compared our technique to a few of the existing methods and further we plan to compare our results with a larger domain of established techniques. We also plan to extend our ideas to develop methodologies for extracting techniques from scientific papers. This shall present challenges as relevant features for technique words may not be readily available, and would require extensive brainstorming.

References

[1] C. Boonthum, S. Toida and I. Levinstein, "Preposition Senses: Generalized Disambiguation Model", *International Conference on Computational Linguistics and Intelligent Text Processing 2006*, Proc. Lecture Notes in Computer Science (LNCS), Springer, pp. 196-207.

[2] V. Srikumar, and D. Roth, "Modeling Semantic Relations Expressed by Prepositions", *Transactions of the Association for Computational Linguistics (ACL)*, vol 1, pp. 231-242, 2013.

[3] S. Lakhanpal, A. Gupta and R. Agrawal, "Towards Extracting Domains from Research Publications", *Modern Artificial Intelligence and Cognitive Science Conference (MAICS) 2015,* Proc. CEUR Workshop Proceedings, vol 1353, pp. 117-120.

[4] S. Gupta and C. Manning, "Analyzing the Dynamics of Research by Extracting Key Aspects of Scientific Papers", *International Joint Conference on Natural Language Processing (IJCNLP) 2011*, Chiang Mai, Thailand Nov 8-13, Proc. ACL pp. 1 – 9.

[5] C. Yan. *Hidden Markov Models* [Online]. Available: http://digital.cs.usu.edu/~cyan/CS7960

[6] X. Rong et al., "Combining Text Classification and Hidden Markov Modeling Techniques for Structuring Randomized Clinical Trial Abstracts", *American Medical Informatics Association (AMIA) Annual Symposium 2006*, Proc. AMIA Annu Symp Proc, pp. 824 – 828.

[7] S. Lakhanpal, A. Gupta and R. Agrawal, "On Discovering Most Frequent Research Trends in a Scientific Discipline using a Text Mining Technique", *52nd Annual ACM Southeast Conference 2014*, Kennesaw, GA, March 28-29, Proc. ACM SE, pp. 52:1-52:4.

[8] Cambridge Dictionaries Online. *Prepositional phrases* [Online]. Available: http://dictionary.cambridge.org/us/grammar/british-grammar/prepositional-phrases

[9] A. Hertzmann. (2010). *Writing Research Paper* [Online]. Available: http://www.dgp.toronto.edu/~hertzman/courses/gradSkills/2010/writing.pdf

[10]BioMed Central. *Writing titles and abstracts* [Online]. Available: http://www.biomedcentral.com/authors/abstracts

[11]S. Isaacs. *Headlines* [Online]. Available: http://www.columbia.edu/itc/journalism/isaacs/client_edit/Headlines.html

[12]*Writing a Scientific Research Paper* [Online]. Available: http://www.yale.edu/graduateschool/writing/forms/Writing%20a%20Scientific%20Research%20Paper.pdf

[13]A. Sherman. (1996). *Some Advice on Writing a Technical Report* [Online]. Available: http://www.csee.umbc.edu/~sherman/Courses/documents/TR_how_to.html

[14]P. Koopman. (1997). *How to Write an Abstract* [Online]. Available: http://users.ece.cmu.edu/~koopman/essays/abstract.html

[15]*Naïve Bayes* [Online]. Available: http://www.cs.colostate.edu/~cs545/fall13/dokuwiki/lib/exe/fetch.php?media=wiki:13_naive_bayes.pdf