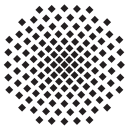

Universität Stuttgart

Institute of Parallel and Distributed Systems

University of Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Master's Thesis Nr. 0202-0002

Conditional Random Fields based Knowledge Retrieval in Mobile Environments

Muzahid Hussain

Course of Study:	Information Technology/InfoTECH
Examiner:	Prof. Dr. Kurt Rothermel
Supervisor:	Dr. Adnan Tariq
	Dipl.-Inf.Thomas Bach
Commenced:	2015-02-02
Completed:	2015-09-25
CR-Classification:	

Abstract

The exponential growth of digital products lead to the massive generation of data every day at the alarming volume,velocity and diversity. In order to extract meaningful value,advanced storage ,processing and analytical systems are growing fast. Collective Adaptive Systems(CAS) are such large scale distributed systems which provide a scalable and efficient way of storing and processing the data at such a massive scale.Allow Ensemble[13] is one such CAS systems where participants(nodes) have the knowledge repository based on probabilistic graphical models.These model are built from the observed data in different contexts. These models are associated with a degree of learning based on the quality and amount of observed data. The participants of CAS are loosely connected components differing in dimensionality of these knowledge repositories in a dynamic environment. The efficient retrieval of knowledge from these probabilistic models possess a greater challenge and is the core task of this thesis work. The major challenges we faced to achieve this task includes: 1.A confidence measure to quantify the degree of learning of the knowledge model on each participant (network node). 2.This confidence measure should answer the query in absolute sense in a way that it determine the confidence in knowledge model wrt to itself when tuned to give the best/saturated knowledge. 3.The mechanism needs to be developed to aggregate this measure for a group of crf nodes to determine the overall average degree of learning. 4.The efficient routing mechanism need to be developed to answer the query with specified confidence measure of learning.

We propose the following concepts to handle this task. 1.Confidence Metric which quantify the degree of learning and providing the absolute sense of learning using the statistical concept of Confidence intervals. The confidence value is associated with every node of the probabilistic crf graph and these values can be aggregated to produce an overall confidence score. 2.We have developed an efficient routing mechanism so that query can be routed to the best learned knowledge model specific to its context parameters. Each participant stores and propagates the information about the knowledge model reachable through its neighbor. We call this approach knowledge aggregation based routing. 3.We have proposed a new routing scheme of query learning in which routing behavior is developed from the past queries.

In general,the high retrieval accuracy is obtained in both scenarios. We believe that our approach of summarization ,propagation and effective retrieval of knowledge in a probabilistic models poses great direction to further enhance and develop interesting applications that can make good use of our research work.

Acknowledgements

I would like to express my sincere gratitude to my supervisors Dr. rer. nat. Muhammad Adnan Tariq and Dipl.-Inf. Thomas Bach for their continuous support during this whole period. Their enthusiasm, patience, motivation, and immense knowledge guided me all the time. I could not have imagined having a "better mentor" than them. I am particularly grateful for the assistance given by Prof. Dr. rer. nat. Dr. h. c. Kurt Rothermel, to provide me the opportunity and resources to do my thesis at the department of distributed systems.My special thanks to the staff of the department for their gentle and prompt support. Last but not the least,I would like to thank my family , without them I could not have walked a single step in my life.

Contents

List of Figures

List of Algorithms

Chapter 1

Introduction

The exponential growth of digital and social media products lead to the massive generation of data everyday.All the digital systems around us from small data sensors[11],smart phones to big super computing devices[16] generates and transmit tons of data at the alarming volume,velocity and diversity. This data is an important tool from which intelligently system and user behavior can be extracted. The collective field of storing,processing and applying analytics on this massive data consists of Big data technology[2]. It is characterized by its three Vs (Velocity,Volume and Variability) of data manipulation which form the core pillars of this technology [2],[16]. In order to extract meaningful value from this data,advanced storage,processing and analytical systems are revolutionized. The science of recognizing patterns in this massive data comes into picture. Data mining and machine learning algorithms helped to extract the hidden meaningful value from this data. There are various applications centered around this emerging field of Big data like social networks[16],cloud computing[17],Internet of things[1],search engines[10].

Along with the advanced hardware capabilities,convectional software algorithms and frameworks are also getting improved. The most obvious and pragmatic trend to handle this massive data is to store,process in a cluster nodes of distributed systems. Distributes Systems provide a very scalable and efficient way of storing and processing Big data[18] but the massive scale poses additional challenges for these processes to get implemented. Along with that,retrieval of this data at such a massive scale need improved and optimized techniques. The clusters generally have very diverse sets of data differing in dimensionality[15].One such category of distributed systems where massive data is handled and manipulated are Collective Adaptive Systems.

Collective Adaptive Systems are the large scale distributed systems where the participants are loosely connected in heterogeneous and dynamic changing environment [15]. Urban mobility Systems in which different autonomous participants form a spontaneous network of information exchange is a good example of Collective Adaptive System. The Allow project under the European Commission research is one such application of CAS[13].The urban mobility scenario is an application of Allow project. The autonomous participants in an urban mobility network are the agents of real world Allow nodes like passengers,buses,trains as well as any computing devices like smart phones and back end route/payment planning systems. Each Allow node/participants should be able to store,represent and retrieve observations from the past experience[13]. These observations are then represented in a compact way through modeling. In Allow research project,probabilistic based modeling called Conditional random fields(crf)[20] are used to model these observations,which then called as knowledge models.

The participants have diverse and different set of knowledge stored in them respective to their context of observations. In such a scenario,participants have different confidence about the learning for different contexts. For example,a person A traveling regularly from Stuttgart to Munich compared to a person B who travels very less in the same route is more confident about the query specific to this route (context) when answering from its local knowledge model developed from the observations collected about this route. Since different observations collected in different contexts lead to high diversity in knowledge models,the scalable retrieval of this stored knowledge poses a very challenging task. In addition,the knowledge in the autonomous participants constantly updated through the influx of incoming observations. The retrieval of knowledge makes such systems quite different compared to the traditional systems of data retrieval. The task involves intelligently interpreting the data and retrieving the knowledge pertaining to data rather than simply retrieving some specific data. Since the knowledge is modeled through conditional random fields,confidence scores of regions of learning need to be retrieved pertaining to certain contexts. These scores can then be used to build routing models which determine the routing path of any query to higher region of learning. In a nutshell, this thesis deals about the efficient retrieval of knowledge stored as probabilistic models for different contexts in a distributed system.

1.1 Contribution

The following are the main contributions of this thesis :

- We have proposed a Confidence Metric which quantify the degree of learning of the knowledge model specific to the context. A higher confidence value means an Allow node can deliver a better knowledge specific to the query parameters. Confidence metric quantify the absolute sense of learning i.e the saturated state of learning achievable of the knowledge model wrt to query. The confidence value is associated with every node of the probabilistic crf graph and these values are then merged to produce an overall confidence score of an Allow node wrt to a query. In order to achieve absolute sense of learning we are exploiting the statistical concept of Confidence intervals. Each crf node has an error distribution obtained from difference between actual and predicted values of model output. Confidence metric on each crf node is created from combing mean error and deviation of this distribution.
- We have developed an efficient routing mechanism so that query can be routed to the best learned knowledge model specific to its context parameters. Each participant keeps the information about the knowledge model reachable through its neighbor. Each Allow node aggregated the information about its crf based knowledge model and forward this aggregated information to its neighbors. We call this approach knowledge aggregation. The aggregation basically involves the clustering of crf graph nodes into regions of similar learning and merged their confidence scores. This approach of routing provides high confidence based retrieval of better learning for large and dynamic networks.
- In order to mitigate high message overhead in the knowledge aggregation based routing approach,we have proposed a new scheme of query learning based routing. In this

method,expensive process of aggregated knowledge propagation is avoided by simply learning the routing behavior from the past queries. Each neighbor give feedback to the source Allow node which is an estimation of the confidence for that query. The source allow node which sent the query build routing model from this feedback to deliver routing decision for future queries. The confidence metric proposed in point I,used as a feedback to the source node wrt to query. Thus,the confidence estimation of any neighbor is based on the feedback collected from the queries sent via the path of this neighbor. In this way,propagation of aggregated knowledge is avoided which was the quite expensive process in the previous approach of routing.

- We have investigated the retrieval accuracy of the two routing approaches proposed and discuss the future direction of this research work.

1.2 Thesis Organization

The structure of the thesis is provided. Chapter 2 is about the related work for this thesis while in Chapter 3,system modeling and problem formulation is discussed. In chapter 4,we proposed the concept of confidence based retrieval. It explores the theory behind the developed metric and how it is fulfilling the task of quantifying the saturated state of learning.Chapter 5 discuss about the current literature for graph clustering and the clustering algorithm used for crf graphs. Chapter 6 deals with proposing the knowledge aggregation based routing approach while chapters 7 deals with the developing and discussing query learning based routing approach. The evaluation for the two routing mechanisms are provided in the chapter 8.We summaries the overall work and proposed the future work for this research in the final chapter.

Chapter 2

Related Work

The goal of this chapter is to give overview of all the important concepts which form the basis of the thesis.In this chapter,we will discuss the core concepts and its related literature used in the thesis.We start with the overview of knowledge retrieval in P2P systems and the general idea about publish subscribe networks. We then discuss probabilistic graphical modeling and one of its special type of undirected graphical model called Conditional Random Fields which forms the baseline of our system. We will then present the concept of confidence intervals used in measuring the quality of learning of our system.The relevant survey behind the various clustering techniques used for knowledge aggregation are discussed in a separate chapter.

2.1 Knowledge Retrieval in P2P Systems

The information retrieval is one of the popular research topics from the last decade especially with the rise of web technologies**??**,Big data,and search engines competitiveness. The scalability and efficiency of retrieval especially in P2P networks which are decentralized and distributed in nature face an additional hurdle[21]. P2P systems exploit vast amount of computational power,memory from small computing nodes in a distributed environment.Every peer can behave as client ans server and they corporate in large numbers to share information and other resources.[27].
Now,we will explore more about the knowledge retrieval and how it is different from the convectional data/information retrieval. Data retrieval includes storage and efficient retrieval of structured data like in DBMS systems. Information retrieval deals with extracting information efficiently and compactly like extracting relevant documents in search engines using indexing methods[27]. Knowledge retrieval systems provide a summarized representation by improvised search and manipulation. In a nutshell rather than representing data or just information in base system(from where retrieval need to be done),knowledge in represented in place of them and efficient mechanism is developed to extract knowledge or retrieval of source of knowledge.For example,We can use probabilistic modeling to create a knowledge model from the observed data rather than simply storing the data in raw form.Then our retrieval task would include summarizing the knowledge model efficiently and developing a good extracting mechanism of this summarized knowledge.
When it comes to retrieval in P2P,the present literature has mainly two directions to proceed. The first direction is about the distributed indexing structures for similarity searches [21]. The other approach involves keeping the indexing on the peer intact and interpret task as

routing problem ie routing the query to that specific peer which can deliver for the search parameters[22]. The knowledge retrieval in P2P systems are yet to explore fully and this thesis work is the extension in the direction of applying routing strategies for data retrieval[13] for probabilistic models.

2.2 Publish Subscribe Networks

Publish Subscribe networks are the types of P2P systems where set of nodes register to events while other set of nodes are register to provide content specific to those events. The communication is event based where peers can subscribe (register) and publish to an event. Thus it will lead to loose coupling in space and time. Peers need not to block or geographically aware about each other to exchange information.[26] The following categories of Pub sub systems are:

- Topic Based Modeling :A subscriber receives events of a particular topic which it is subscribed for. Individual filter cannot be specified so it is courser grain selection.
- Content Based Model : Depending on the properties of the events,subscribers can mention the filtering conditions.
- Concept Based Modeling : The filtering on event is done on concepts rather than on attributes values in the absence of definite information about event i.e when syntactic and semantic structure of an event is unknown.

Further,the routing algorithms for the pub sub networks can broadly be dividing into the following.

- Event Flooding: Events are broadcast and every node can decide locally if this event matches a subscription.
- Subscription Flooding: Each publisher sends only the subscribed event to the node when these nodes flood their subscriptions in the network. Again,it will lead to large overhead and does not provide a scalable routing mechanism.
- Filtering Based Routing: In this type of routing,an event is routed only to interested subscribers rather than flooded across the network. The routing tables are formed which contains the routing information about the neighbor. Thus,it will lead to publish out only subscriptions in which neighbors are subscribed.

Routing deals of migrating the queries/events to the content providers[14]. In our approach of knowledge aggregation,we have used filtering based routing approach where routing information about every neighbor is kept on every node in an aggregated form [14] [13]. The knowledge contents on node are aggregated and propagated to its neighbors so that every node in future knows about its neighbors subscription or the information it can deliver to its neighbors.

2.3 Probabilistic Graphical Models

It is quite frequent in real world that dependencies exist among the different data objects and it has to be taken into account while create the data model. For example in the classification of web documents,page text provides information about the document class label and hyper links are the relations among the documents. If we only take the text into account and not the links,we will lose much information which is present due to the dependencies among various documents through hyperlinks [43].The good model should incorporate both sources of information and it should be simple enough to clearly state and depict the real world. The graphical models are the formalism to represent all these dependencies among the data objects precisely and clearly.
Definition:Probabilistic graphical models are the representation of probability distribution **P(Y,X)** of a set of random variables by a graphical structure along with their parameterization.
The variables Y are the attributes of the object we want to predict and X are the observed knowledge about the object[7]. In general sense,Y are the output of the model and Y its input. The various parts of graphical structure (cliques of a graph) are associated by parameters and these parameters provide the values of probability distribution. Thus,PGM is a comprehensive framework to represent and compute complex probability distribution [40].The PGM has numerous important applications in the field of engineering like part of speech tagging [20],information extraction[23],robot motion estimation[43] etc.
A graphical model is a family of distributions that factorizes respecting the underlying structure of the graph[7].Since these graphs are really large,in order to achieve computational feasibility,we divide these distributions in a product of local functions which combined only a small set of random variables. These variables follow Markov property so division into local functions is possible. These local functions are called factors of a graph. Thus,probability distribution of any graphical model can be written as the product of its factors for any combinations of factors(called local functions) Ψ_c.

$$P(\vec{v}) = 1/Z\Pi_c \in_C \Psi_c(\vec{v}) \tag{2.1}$$

These factors are the probability for the cliques(clusters) of a graph and overall probability is the product of these individual probabilities over various cliques of a graph.The Z is simply normalization factor and is constant. It is defined as partition function and it basically bounds the distribution sum to 1.
PGM are of two categories,directed graphical models which are called Bayesian Networks and undirected models which are called Markov Random Fields [43].Markov Fields are undirected graphical representation between random variables where each random variable follows the Markov property. The property states that random variable is conditional independent of all the other variables in the model provided its neighbors are observed. The most important class of Markov networks is Conditional random fields.Traditionally,graphical models the joint probability distribution but for a large network and dependencies among relational data,it is very computationally challenging as it requires to model the input variables distribution P. Lafferty [43] presented a novel solution to this problem by directly modeling the conditional distribution of **P(Y/X)** considering observed knowledge X as non-random variable. Thus,crf

is a conditional distribution of an associated graphical structure with some observed knowledge as input[43]. Now we will explore this undirected class of graphical model which constituent the basic framework of our system.

2.4 Conditional Random Fields

In the previous topic,graphical models as a diagrammatic representation of probability distribution is discussed. The idea of conditional probabilistic model is presented which is used in modeling knowledge model of our system. In this section the theoretical foundation of the Conditional random fields is build. Let G=(V,E) be the graph where V are the set of random variables and E are the connecting egdes between them. This set can be further written as V = XUY,where X is the set of input variables that are observed and Y is the set of output variables that we want to predict of the system model. X can take values from set X and similarly Y can take values from Y. We generally call these values of Y as class labels since we want to predict these labels for every random variable of the model.This graphical structure formed as underlying framework to represent and compute Conditional probability distribution of CRF.

Definition: Conditional random fields are the probabilistic models satisfying the underlying graphical structure G=(V,E) to compute the conditional probability of $P(\vec{y}/\vec{x})$ of a predictable output$\vec{y} = (y_1...y_n)$ given the observed input $\vec{x} = (x_1...x_n)$. Conditional Probability $P(\vec{y}/\vec{x})$ can further be written as

$$P(\vec{y}/\vec{x}) = P(\vec{y},\vec{x})/P(\vec{x}) \tag{2.2}$$

From general equation of graphical model,it can be written as

$$P(\vec{y}/\vec{x}) = 1/Z * \Pi_c \in_C \Psi_c(\vec{y},\vec{x})) \tag{2.3}$$

Conditional Random fields can be represented in the form of log linear model. Log linear model is a mathematical representation as a function whose log is a linear combination of the parameters of the model[43]. That is,it has the general form

$$\exp(c + \sum_{i=1}^{m} \lambda_i * f_i(\vec{y},\vec{x})) \tag{2.4}$$

where $f_i(\vec{y},\vec{x})$ are the functions of X and Y (called as features)and λ_i are the model parameters. Training data(where input(X) are its corresponding class labels outputs(Y) are provided) can be represented by features or indicators functions of a log linear model. For example

$$f_i(x,y) = 1, if y = 5 and x = Sunny 0, otherwise \tag{2.5}$$

$$\tag{2.6}$$

Merging I and II,factors of a crf can be expressed as

$$\Psi_j(\vec{y},\vec{x}) = \exp(\sum_{i=1}^{m} \lambda_i * f_i(y_j, y_{j-1}, j, \vec{x})) \tag{2.7}$$

$$\text{where} f_i(y_j, y_{j-1}, j, \vec{x}) \tag{2.8}$$

is a feature depends on previous and current label values,input observation vector and current sequence indicator j. Thus,Conditional Probability of crf can be written as

$$P(\vec{y}/\vec{x}) = 1/Z * \exp(\sum_{j=1}^{n}\sum_{i=1}^{m} \lambda_i * f_i(y_j, y_{j-1}, j, \vec{x})) \tag{2.9}$$

The two most important concepts in CRF models can be understandable easily by these two problems.

- "Given observation x and a CRF M,How do you find the most probably fitting label sequence y" ?
- "Given label sequence Y and observations,How to find parameters of CRF M to maximize" ?

While problem I in CRF modeling is defined as Inference and in fact the most common application of CRF model,problem II is training or learning of CRF model to get best parameters of the model. Now we further elaborate these two important aspects of CRF modeling[40].

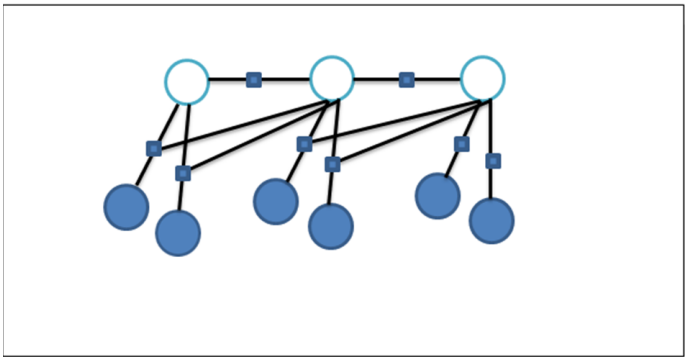

Figure 2.1: Conditional Random Fields

2.4.1 Learning

The learning of the CRF model is actually the parameter λ_i estimation to determine the optimized values such that it best fit the probabilistic model. The general method for training /learning in probabilistic models is maximum likelihood estimation. The method states that the training of model is done by maximizing the log like hood function on the training data.We will not go into the details of likelihood function but for an overview : "The likelihood of a

set of parameter values,λ,given data Training data T,is equal to the probability of those observed/training data given those parameter values,"[] that is

$$L(\lambda/T) = P(T/\lambda) \tag{2.10}$$

The basic concept involves the convex optimization of the likelihood function and the values of parameters λ are considered as final learned parameters at the instance of maximization.

2.4.2 Inference

The inference in sequential probabilistic models like CRF is the problem of finding the most likely sequence of Y for the given observations X.(Rabiner 1989)[40]. It is a NP hard problem but if the graph structure does not contain loops,there are exact inference approaches in the literature[7]. In other cases,we calculate the approximate inference. Generally,in order to solve this dynamic programming challenge of statistical inference,Viterbi Algorithm is used[43]. The query at any Allow node is answered by the CRF inference. i.e the query input parameters contain the observation about the system and most likely sequence provided by the CRF inference interface is the output of the model against certain query.

2.5 Confidence Interval

In this section,the statistical concept of Confidence Interval is described thoroughly. Confidence Interval is the world of statistics is an estimate of an interval of a population parameter.The goal is to estimate the population parameters provided the observed data. “A confidence interval gives an estimated range of values which is likely to include an unknown population parameter,the estimated range being calculated from a given set of sample data” [5]. CI gives the range where the parameter value can lie in the population.
Confidence level determines how frequently the interval contains the parameter of interest[3]. Confidence intervals are framed at certain confidence level say 95 percent. It means that if the population is sampled and estimates of observed intervals are being made on every event,the true population parameter would be bounded in the resulting intervals in approx. 95 percent of the events. Let the mean obtained from the sample represents the mean of the whole population. Are the random sample contains enough number of observations so that the mean obtained actually is the true mean ? The basic goal of approximating the true value of population parameter is addressed by CI in a way it provides a range of values which will be mostly likely to contain the parameter of interest.
Definition : Confidence Interval :“For a population with unknown mean μ and known standard deviation σ,a confidence interval for the population mean,based on a simple random sample (SRS) of size n,is $\mu + z * \left(\frac{\sigma}{\sqrt{n}}\right)$,where z* is the upper (1-C)/2 critical value for the standard normal distribution”.
Margin of Error ($z * \left(\frac{\sigma}{\sqrt{n}}\right)$) is the value associated with population parameter which determines the width of CI [5]. Fig 2.2 shows probability density area curve where for a CI with a confidence level C,area in each tail is 1-C/2.It shows that CI with level C,probability value

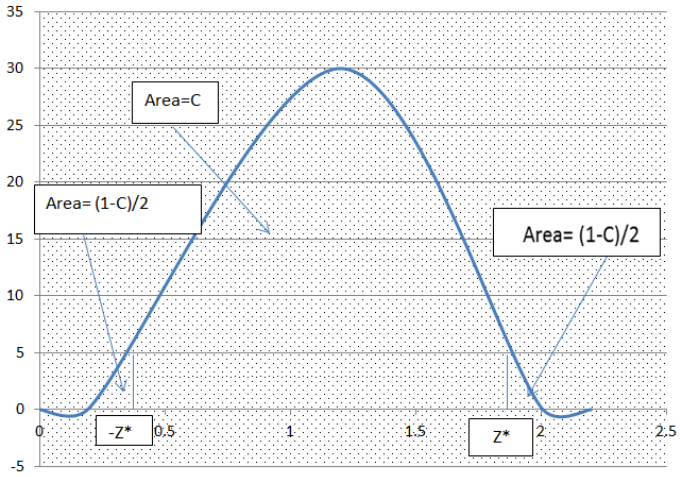

Figure 2.2: Confidence Interval

is 1-C/2.It comes from the z curve in which z represents the point on curve such that P(Z>z) = p = (1-C/2),for 95% CI,the value of z =1.96. As the sample size increases,the size of CI will decrease without reducing the confidence level. It is because of the fact that effective standard deviation decreases as sample size increases.

It is necessary to remind that larger value of confidence gives wider interval i.e. 99% confidence intervals are wider than 95% of confidence intervals. It make sense when we try to understand confidence as degree that our interval contains the true parameter. Lets say user want to be 100% sure that every time calculated interval contains the true parameter,interval must contain every possible value and thus it will be much wider. When it is acceptable that only 50% times interval contains the true parameter,interval will be much narrower.

In a nutshell,we have discuss the majority of background knowledge required to investigate our problem task and drawing its potential solutions. Now,we will move to the chapter of System Modeling and problem formulation which explores the major challenges and the system model.

Chapter 3

System Modeling and Problem Analysis

In this chapter,we will describes the distributed network of Allow Ensembles with its participants and their corresponding functionality. Then we will explain the Evolutionary knowledge model along with the idea of developing quality measure for quantifying learning of the model. We will cover the current problem in detail and explore its potential solutions in summarizing and routing the information in a distributed environment.

3.1 Network

We have a distributed system of nodes p_1,p_2,p_3,p_n communicating to each other in a dynamic mobile environment. The communication can be direct or indirect depending upon the connections. The links are not static in a sense that both node and links can go down anytime in a dynamic scenario like in a mobile environment. From now onwards we call this node as Allow node. An Allow node in simple sense can be considered as a device (eg smart phone) which runs a communication protocol[12].There is not an upper bound in the delivery of messages and process execution. Random delays and duplicated messages are quite possible in such an environment.
Each Allow node has a repository defined as Knowledge model which is the representation of the knowledge build from the observations. Now,we move on to discuss these individual participants of the system.

3.2 System Model

3.2.1 Knowledge Model

Evolutionary Knowledge(knowledge model) is defined as an intelligent and flexible knowledge repository which is used to model the behavior of an Allow node[12]. It is a model learned to predict parameters of the system and their dependencies. It can be considered as a repository where application specific parameters are stored and represented efficiently such that they can used to answer any query in the future.Conditional random fields are used as a framework to

build the knowledge model.
We try to comprehend the concepts applying Evolutionary Knowledge model on the urban mobility scenario in a city.
The urban mobility is a complex transport system of a region.For sake of understanding,consider it as a bus system of the region.We have to build a system which can provide travel time between specified bus stations when queried by the user. The network Allow node can be any computing device like smart phone and possessed by the user. The knowledge model is a probabilistic graph. We can model this real world where road segments between two stations can be considered as nodes of our knowledge model and their dependencies between segments modeled as edges.The knowledge model is a probabilistic framework which takes input arguments(observed/context parameters) provided from the user query like time of day,weather conditions and provides total travel time as output.This model is an evolving model which gives some predictions for the given query parameters.It then learn from these scenarios to evolve and to better answer a query in the future.
Each Allow node has a local knowledge model which is formed from the local learning and some information about remote knowledge models through some efficient representation.Fig 3.1 shows the knowledge model of the urban utility where observed parameters are Time of day and Weather (marked in red) and output variable is travel time. All these nodes take discrete set of values as a range.
Now,consider a scenario of some users who has different frequency of visiting the same region of bus network.Clearly,the user has a better understanding of his/her own view of the transportation system since more learning is done for that particular view.The user possesses an Allow node (smartphone) which has local Knowledge Model as well as the some knowledge of the other Allow nodes of the network. The user query the total travel time between two stations A and B providing context(input) parameters,the best possible prediction of travel time is expected for this query.
Our system should be intelligent enough to comprehend user constraints and preferences to create the most optimized solution. For an example,the query from the user can be to provide total travel time between station A and B considering the global constraints/context that time of the day is afternoon and weather is rainy. The additional provided parameters can be taken as user constraints or preferences wrt to query. Lets understand it with a help of an example. In the urban mobility scenario,the person I uses bus service to go to his office near the city center. Its local knowledge model learns the travel time of the path during different contexts. These contexts are set of observations like time of day and weather conditions. Moreover there is a person II who visits city center once in a month. The local knowledge model of person I certainly contains much more knowledge about the city center travel time as compared to the model of person II. Now if another person III wants to know the travel time to city center during different weather conditions /time of day,person I local Knowledge model can answer this query very well as compared to person II.It is due to the fact that since it is very similar to the knowledge learned by Knowledge model of person I.

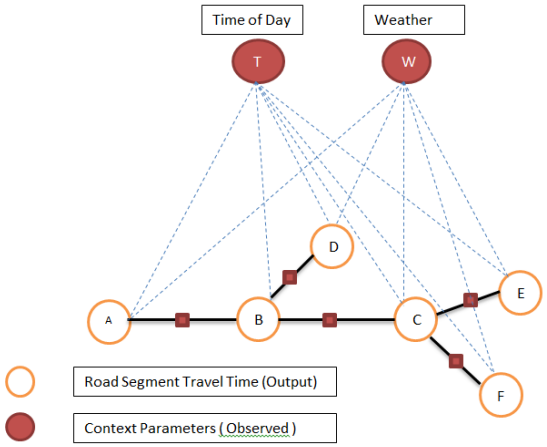

Figure 3.1: Knowledge Model

3.2.2 Conditional Random Fields as framework for Knowledge Model

As described previously,CRF are the probabilistic graphical approach of modeling systems where system elements are represented using random variables(RV) and their interactions are modeled by the edges between them.
Let the graph G = (V,E) consists of nodes V and edges E.Let O be the set of all observed variables and Y be the set of all random variables whose state is to be determined. Moreover,let L be the set of states/labels which every random variable can take. In CRF model,every RV is represented by one template object which changes the transitions from one state to another and these transitions are represented by the edges of the graph. The set of edges are represented as E in graph which depicts the interaction (positive or negative) between corresponding nodes.

In our example of transport network shown in Fig 3.1,each node of conditional random field is modeled as road segment and these nodes are connected by edges to show the interaction of road segments on one another.For an instance,edge between road segment of station B and station C says that time taken to clear this segment depends upon the time taken by clearing previous road segment.The model is associated with reliability at node level that takes an account of number of instances used to train that Allow node of model.
The quality measure determines the quality/degree of learning of the knowledge model in an absolute sense i.e. the quality achieved till the current learning to that of the best/saturated state of the learned model. The node/edge is the minimum granularity level and idea is to merge these quality measures associated to form clusters of same learning.

Confidence in Knowledge

The Allow nodes in the network have knowledge models and any query should ideally be satisfied by the globally best output data among the whole network. As already exposed in

previous sections,bandwidth is one of the most critical resources to consider in a distributed environment. There can be different approaches to answer any query in the system,one naive approach can be to flood the query in the network and let the Allow node which has the best learned model answer it. In our system where there can be million on Allow nodes (distributed nodes),this would lead to very high usage of bandwidth resource. In order to answer a query in such a scenario,we need an absolute quality of learning of knowledge model which gives certain probabilistic score that the Allow node can answer the specified query efficiently.
Conditional Random fields constituent the Knowledge model. Although the template of model is same on every Allow node,a crf node can have different quality of learning on different Allow nodes. Thus,each crf node in the model is associated with confidence which depicts its quality of learning.Moreover,this confidence measure must give an absolute idea about the current quality of learning of the crf node wrt query. If the quality conditions expected by the query are met on the local Allow node by the learned model so far,this Allow node must answered the query rather than propagating to neighbor Allow nodes.
We are using the statistical concept of confidence intervals to depict the degree of learning achieved so far wrt to the best learning that can be achieved.
"For a population with unknown mean μ and known standard deviation σ,a confidence interval for the population mean,based on a simple random sample (SRS) of size n,is $\mu + z* \left(\frac{\sigma}{\sqrt{n}}\right)$,where z* is the upper critical value for the standard normal distribution".
Margin of Error ($z * \left(\frac{\sigma}{\sqrt{n}}\right)$) is the value associated with population parameter which determines the width of CI.(ref to Fig 2.2)
The probability distribution of error is obtained using a feedback loop on predicted values and true values for each crf nodes of the model. The mean of this error distribution give the idea of fit of the model or one can say the degree of learning. Along with that,the margin of error(deviation from mean) obtained from confidence Intervals give the absolute sense of learning wrt to the model's saturated state of learning.The lesser the margin of error,reliable the model is to answer the query since it can be interpreted from margin of error that the model has enough learning on sufficient amount of quality data.

3.3 Query Composition

The input query to the network can be initiated at any Allow node. We adhere to our classical example of urban mobility so a sample query for the set scenarios would comprise of the input arguments as weather,time of the day,list of road segments for which travel time is needed as output.Along with that,the minimum confidence required is alos provided as input parameter. In order to implement,this query can be comprehended with the following parameters.

- Minimum Confidence($C_{\min}$) :It is the minimum confidence value which must be satisfied by the answering Knowledge model at any Allow node in the network.
- List of crf nodes to inspect($List < crfNodeId >$) :Suppose the query comprises of the path which can be broken into list of road segments/blocks for which total travel time is need to be calculated. Since the road segment is mapped as crf node in the probabilistic crf model,these road segments are mapped to the crf nodes in the background crf graph

modeling the real world road network.Thus,it is considered as a list of crf nodes with their node Id as element.

- Weather(W) : It is an observed parameter for the crf model which is a discrete variable having Sunny,Rainy,Snow,Cloudy as domain values.
- Time of the day(T) : Like the previous parameter,is a global observed variable for the crf model,discrete in nature with Morning,Afternoon,Evening,Night as domain values.

3.4 Query Routing

In Fig 3.2,a sample topology with Allow nodes having knowledge models are provided. A query with a threshold minimum confidence is received on Allow node X but it finds that it do not have enough knowledge to answer this query with a given confidence. In that case,query is propagated across the network through the use of efficient routing mechanism until it reaches a Allow node which can answer it with enough confidence.
In a nutshell,

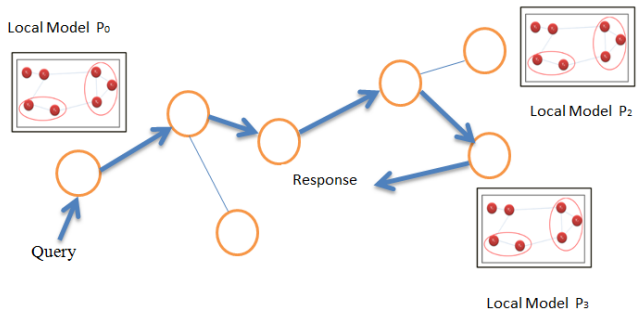

Figure 3.2: Query Routing

Given: A query Q(arg,C_{min}) is provided on Allow node X, C_{min} is the measure of knowledge expected specific to input arguments.
Find: In case local Allow node does not have enough confidence to answer the query,it will be routed. Any neighbor which can answer with at least threshold confidence will answer this query. The efficient routing mechanism supports the overall process to reach the query on an appropriate Allow node.
Every Allow node has a local knowledge model along with routing information about its remote knowledge models which will help to route the query. The major challenge is to develop the scalable routing mechanism to reach out the best fit remote Allow node which can answer the query with a given threshold confidence.
To summarize,the main challenges in this thesis cover the following points:

- A measure to quantify the degree of learning of the knowledge model on each Allow node wrt to the specified query has to be determined.
- This measure of learning of the knowledge model should answer the query in absolute sense in a way that it determine the confidence in knowledge model wrt to itself when tuned to give the best/saturated knowledge.
- For scalability,the mechanism needs to be developed to aggregate this measure for a group of crf nodes to determine the overall average degree of learning in a group of crf nodes.
- The efficient routing mechanism need to be developed to answer the query with specified measure of learning.

Chapter 4

Confidence Based Retrieval

4.1 Concept of Confidence

The need and significance of Confidence Metric for Knowledge management has been briefly introduced in the chapter of system modeling.
Any query must be satisfied by the best learned model among the whole network of Allow nodes with diverse knowledge models. A naive approach can be to flood the query in the network and let the Allow node which has the best learned model for this query to handle the query. In our system where there can be millions of Allow nodes,this would lead to very high usage of bandwidth resource. In order to handle a query in such a scenario,we need an absolute quality of measure of a learning model. This quality gives certain probabilistic score which quantify the degree of learning of a Allow node.
In the classical machine learning approach,the degree or quality of learning of a model depends upon lots of factors where some important ones include the quality and amount of training data along with the number of features used. The quality measure should give some probabilistic score to determine the degree by which the specified Allow node can answer the query. This score defines our Confidence metric which reflect the quality of learning of the knowledge model.
Confidence metric depends upon the learning parameters of the model as well as the query parameters. As stated before,the query parameters will define the scope of the query and our Confidence metric provide the quality score of learned knowledge model corresponding to the scope(group of crf nodes) of query. If the system has very little or no knowledge within the specified scope of the query,it should get reflected to the user by the Confidence score. Thus,system provides the best predicted output model satisfying the minimum query's confidence score and scope provided by the user.
In a nutshell,Confidence is the quality measure which quantifies the degree of learning of the knowledge model with respect to the specified query.

4.2 Quantifying Confidence

We will now develop Confidence function satisfying the requirements imposed in the above mentioned literature. As discussed in chapter 2,we have explored many statistical techniques

to determine the confidence metric. Confidence score should take into account of the saturation of learning of the model. We want to measure the reliability of the learning as well i.e. on how much data the model is learned upon and whether it has achieved a state by the given data for decision making.
Scope is the area of interest or the sub set of nodes of the knowledge model.There can be any combination or collection of nodes to form a scope provided there is a path between them. Scope contains confidence Interval information (mean and margin of error distribution) and list of crf node in the scope. Each Allow node has a crf based local knowledge model that consists of graph G = (V,E) where V is the set of model nodes and E is the set of connecting edges between them. The Allow node maps the scope of the query with its knowledge model and generate confidence score.
Along with handling the query at knowledge model of an Allow node,Confidence score is also the measure to compare quality of knowledge models of neighboring Allow nodes. Higher confidence score means high quality of learning and a better model and it helps in decision making while route the query to neighboring Allow nodes in case local knowledge cannot have enough confidence to answer it.

4.2.1 Confidence Intervals

In this section,the statistical concept of Confidence Interval is described thoroughly and how it is needed to be incorporated in our Confidence of knowledge model to get the absolute sense of learning.
The goal is to estimate the population parameters provided the observed data. A confidence interval(CI) provides an estimated range of values which has high probability of containing the investigated parameter. CI gives the range where the parameter value can lie in the population.
Definition:
Confidence Interval :"For a population with unknown mean μ and known standard deviation σ,a confidence interval for the population mean,based on a simple random sample (SRS) of size n,is $+ \mu + z * \left(\frac{\sigma}{\sqrt{n}}\right)$,where z* is the upper (1-C)/2 critical value for the standard normal distribution".
Margin of Error ($z * \left(\frac{\sigma}{\sqrt{n}}\right)$) is the value associated with population parameter which determines the width of CI [5].

4.2.2 Node Error

In order to measure the degree of confidence associated with given a node in Knowledge model,the concept of Confidence Interval has been incorporated.Confidence Interval in the world of statistics is an estimate of an interval of a population parameter.The Confidence level associated with an Interval(say 95 percent) gives the chances(probability) that this interval produced has the true value of the parameter of interest.It means that if the population is sampled and estimation of observed intervals are being made on every event,the true population parameter would be bounded in the resulting intervals in approx. 95 percent of the

events.
The notion of Confidence Intervals provides a measure to estimate the population parameter from the sample distribution. This concept can be incorporated in our estimation of quality of model learning. We need some measure of confidence at any instance of learning to check how good our model is so far. Ideally,our system assume that after certain degree of learning,its quality get almost saturated and it can be considered as best to answer the specific query. This quality of learning is associated with views or scopes. For example,a user X who travel between station A and B everyday has much better quality of learning in that scope compared to a user Y which has a traveling frequency of a month in that same scope. Thus,user X local knowledge model has much higher quality of learning in that scope and if a test query is made for that scope of A and B,user X knwoldege model should answer it. The model which is continuously learning must be associated with certain measure of its quality of learning. Moreover,the level of granularity is a node in the model,we want to associate this measure with every node. We define it as Node error as the difference between the true value of the label and predicted from the model. Thus,it is the node accuracy and thus it gives a quality of learning achieved so far from the specific model.
In CRF learning,optimized values of λ are estimated from the function.

$$P(\vec{y}/\vec{x}) = 1/Z * \exp(\sum_{j=1}^{n}\sum_{i=1}^{m} \lambda_i * f_i(y_j, y_{j-1}, j, \vec{x})) \tag{4.1}$$

The significant values of λ are achieved when we will have saturated quality of learning. We have to find a measure to check whether that significance level has been achieved with respect to answering the query. The ideal or exact values after full population observation is taken as target values of parameters and its corresponding function P (Y|X) as target function. This is something an ideal case. At any point of time in model learning,we have a hypothesis function obtained so far from learned parameters which is denoted by h.
Definition: Node error is defined as difference between output values from target function (i.e. true value of labels) and label values predicted from the current hypothesis function h.
$error_s(h) = 1/n * \sum \delta(T \neq h(x))$
Our quality of learning is associated with how well node error from observed samples can estimate the true value of target model distribution node error.It is the node error (misclassification rate) from sampling observations achieved so far.Node Error for the distribution of population (ideal target model) is $error_d(h)$. In general, $N\%$ of area of curve(probability) lies in $\mu + z_N * \sigma$. Combining the notion of CI,it can be said if there is N% Confidence level,then there is N% probability that node error of ideal target model lies in

$$\tag{4.2}$$

$$error_r(h) \pm z_N \sqrt{error_r(h) * (1 - error_r(h))/n} \tag{4.3}$$

$$\text{or approximately,} \tag{4.4}$$

$$error_c(h) \pm z_N \sqrt{error_c(h) * (1 - error_c(h))/n} \tag{4.5}$$

where,$z_N \sqrt{error_c(h) * (1 - error_c(h))/n}$ is the margin of error.
In this way,we can obtained the error distribution of population(ideal case when best learning

is achieved) from the sample error distribution with some degree of uncertainty i.e. the interval that the true error can vary from sample error.

Weighted Node Error

In the previous section,we defined Node Error as simple misclassification rate but it can lead to very high error rate when considering with measuring the total travel time. Instead,we will modify and use weighted Node Error to get more approximate error. $error_s(h) = 1/n * \sum \delta(T \neq h(x))$

4.3 Clustering of Confidence

In this section,the idea of combining the confidence measure for certain number of crf nodes on a Allow node is presented. The knowledge models can contain huge number of crf nodes and it is bandwidth inefficient to forward confidence score of each node of the model in the whole network. We decide to aggregate the confidence scores of set of crf nodes into clusters(called scopes). These clusters are represented by the information of the error distribution(shown in Fig 4.1).

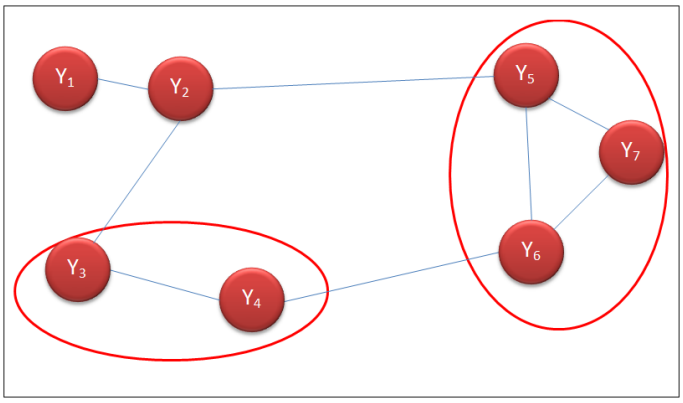

Figure 4.1: Clustering of confidence of crf nodes in Knowledge Model

As we have discussed earlier,each query has a set of nodes as input or the area of interest which it wanted to get the predicted output,we call this area of interest/set of nodes as its Scope. In general Scope is subset of nodes; it can associate with a query or a cluster of CRf nodes of knowledge model. For a query,the set of nodes for which prediction is demanded by the query makes its Scope. The set of nodes in the cluster of knowledge model is model scope.
Definition: Scope is a subset of nodes or the area of interest in the model. It can be associated with query as well as cluster.
The confidence score is associated with each node and we aggregated these scores based on the degree of similarity of their error distributions. We have used a standard approach of

combining two normal distributions(as CI are represented using normal distribution) and along with that the combined confidence interval as well as mean can be combined.

$$C_1 = N\left(\frac{\mu_1, \sigma_1^2}{n_1}\right) \tag{4.6}$$

$$C_2 = N\left(\frac{\mu_2, \sigma_2^2}{n_2}\right) \tag{4.7}$$

Using Standard Error/Margin of Error for CI,it can be constructed as

$$C_1 = N\left(\mu, SE1\right) \tag{4.8}$$

$$C_2 = N\left(\mu, SE2\right) \tag{4.9}$$

$$\tag{4.10}$$

A pooled estimated of the two would be

$$C_e = \left(\frac{n_1 C_1 + n_2 C_2}{n_1 + n_2}\right) \tag{4.11}$$

$$C_e = N\left(\frac{\mu, \sigma^2}{n_1 + n_2}\right) \tag{4.12}$$

$$\tag{4.13}$$

The confidence interval and mean of the cluster represents the effective quality of learning of the clustered nodes (cluster scope). In this way the clusters are created in local knowledge model which contains the groups of aggregated knowledge.

4.4 Confidence Metric

So far we have discussed about the absolute degree of quality which can be derived from the concept of confidence intervals. In a nutshell,we obtained two important measures Node mean error and its associated Standard deviation(margin). Thus each crf node is associated with the confidence intervals parameters (mean and deviation). Our main focus is to summarize this confidence to propagate it in the distributed environment. Now we develop the single value of the confidence obtained from these parameters of confidence intervals.
In order to make a routing decision,at each Allow node we have to decide which neighboring node has the best quality of learning specific to query and then propagate query to it. The quality of learning is available in routing models for the neighboring nodes in the form of aggregated information of confidence intervals parameters. Thus,each routing model has an error distribution with some mean and deviation aggregated with our intelligent clustering technique.

There can be various possible scenarios describing the extremes or boundary points of error distributions for the neighboring nodes. We prefer a distribution which has least error mean and deviation i.e. both tending to zero.The worst distribution would be very high mean and very low deviation since in that we are very certain that error is very high. In a nutshell,we prefer to have a least mean as well as deviation. Thus,if we describe the quality metric combine from the two confidence parameters in such a way that it tends to increase with the decrease of both mean and deviation.

$$Confidence = \sqrt{(1-\mu)} * (1-\sigma) \tag{4.14}$$

Since the quantity (1- μ) and (1 - σ) both tend to increase with the decrease of μ and σ,the value of C will tend to increase. Moreover,(1- μ) lies between zero and one and sqrt function is an increasing function in this interval. We have used sqrt(1- mean) to give preference to the pair which has lower mean compared to lower deviation in a situation such that the quantity (1- μ)* (1- σ) give same value. We could have used the quantity (1- μ)* (1- σ) but since for certain combination (1- μ1)* (1- σ1) = (1- μ2)* (1- σ2),we want to give higher preference to lower mean compared to lower deviation,we used sqrt for mean quantity so that its Confidence value get further higher(shown in Fig.4.2)

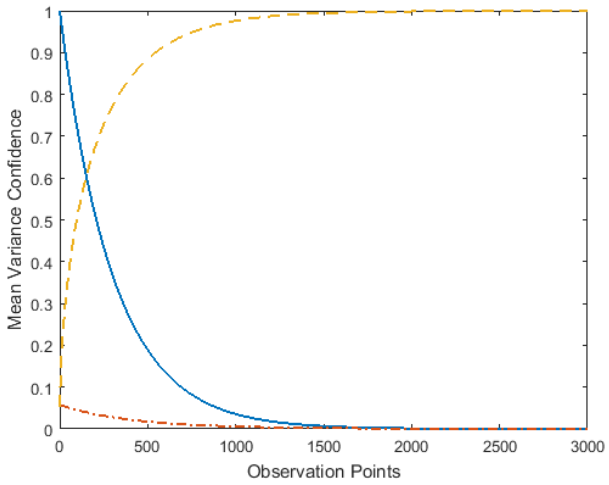

Figure 4.2: Confidence Metric(Mean(Blue),Deviation (Orange),Confidence(Yellow))

4.5 Implications of Confidence Metric

We have taken a confidence metric as a combination of mean and deviation. It will behave according to the Fig.4.2,the behavior of the mean and deviation with change in learning. Although,the metric is chosen in such a way it is satisfying our need,it can certainly vary

accordingly to the need and preferences of the user to alter the behavior of the system. The philosophical aspect in choosing the mean and deviation can leads to entirely different behavior of the system. There are certainly many choices available like low confidence in a small error (case when mean is low but deviation is large) or high confidence in a large error (higher mean but small deviation). For example a user can give preference to the certain distribution accordingly to the dynamics of the network. He may prefer high mean but also high variability (basically high probability if variation) rather than bit low mean and low variability. When we dig more into its philosophical aspects of choosing the appropriate distribution,we realized it has to be explored by digging deep into the field of game theory.
It deals with the task of risk analysis. Let's take it from the perspective of two different strategies of a game. We can assume rather than it is a mere error distribution of a measurement,these are reward distributions of two different strategies of the same game. So now the question rephrases,which strategy would the user prefer,strategy one which is giving a high expected reward with quite a large variance or a strategy two giving a lower expected reward with relatively smaller variance. The decision completely depends upon the risk taking attitude of the user,so there is no simple answer to combine mean and variance to have one metric of quality.

4.6 Distance Metric(Bhattacharya Distance)

In the previous section,we have discussed about the merging of error distributions for knowledge aggregation.The idea is to merge the error distributions of various Crf nodes to form groups having similar degree of learning.
The similarity between two crf nodes are taken by the degree of similarity between their error distributions (assuming normal distribution). In order to extract the similarity of error distributions between two crf nodes,we are using a distance metric which is used in statistics for finding the similarity of two probability distributions [36]. Let p and q represent two multinomial populations each having N classes in their distribution. The respective probabilities would be like p(i=1),p(i=2)...p(i=N) and q(i=1),q(i=2)...q(i=N). Since p(i),q(i) represent probability distributions $\sum_{i=1}^{N} p(i) = 1$ and $\sum_{i=1}^{N} q(i) = 1$. Bhattacharya measure is a divergence type distribution defined by: $BM(p,q) = \sum_{i=1}^{N} \sqrt[2]{p(i).q(i)}$. The geometric interpretation of a measure can be simply taken as the cosine of the angle between N dimensional vectors $(\sqrt{p(1)}, \sqrt{p(2)}., .\sqrt{p(N)})$ and $(\sqrt{q(1)}, \sqrt{q(2)}., .\sqrt{q(N)})$. When the two distributions are identical,

$$\cos(\theta) = \sum_{i=1}^{N} \sqrt[2]{p(i).q(i)} = \sum_{i=1}^{N} p(i) = 1. \tag{4.15}$$

Definition: Bhattacharya Distance is the measure used to find the similarity between two discrete / continuous probability distributions [36]. For domain X,the Bhattacharya distance for continuous distributions p and q is correlated by the Bhattacharya measure(BM) by

$$D_B(p,q) = -\ln(BM(p,q))[36] \tag{4.16}$$

Accordingly,$0 < BM < 1$ and $0 < D_B < \inf$

$$D_B(p,q) = 1/4\ln(1/4(\sigma_p^2/\sigma_q^2 + \sigma_q^2/\sigma_p^2 + 2)) + 1/4((\mu_p - \mu_q)^2/(\sigma_p^2 + \sigma_q^2))[30] \tag{4.17}$$

where,
$D_B(p,q)$ is the Bhattacharyya distance between p and q distributions,
σ_p is the variance of the p-th distribution,
μ_p is the mean of the p-th distribution,
p,q are probability distributions.

Now,we will move into the new aspect of our task which discuss the clustering of knowledge i.e how individual confidence metrics developed which represents the degree of learning can be clustered together.

Chapter 5

Clustering

As we have discussed in the previous sections,crf nodes are associated with error distribution (mean and deviation) which is the measure of their respective quality of learning. Now we have to partition the crf graph into clusters of crf nodes which are combined depending upon the degree of their similarity. The similarity between two crf nodes are taken by the degree of the similarity between their error distributions (assumed normal distribution).Thus,we are aimed to combine crf nodes having similar quality of learning. Now,the similarity of the two normal error distributions is calculated by a distance metric called Bhattacharyya distance[],which has been discussed in the section (SSS). In this chapter,we will explore the various graph clustering techniques employed in order to accomplish our task of clustering similar crf nodes in the graph. The various clustering algorithms are discussed along with the bottlenecks we faced while deploying them for our task.Finally,we have used minimum spanning tree based algorithm to accomplish our task of crf graph clustering.The last section explains the algorithm and how it turned out to be quite suitable for our task.

Definition: The clustering of a graph is the grouping of its vertices taking into consideration of the connectivity between those vertices by the respective connecting edges.
The various methods examined for crf graph clustering are the following.

- K-Nearest Neighbor clustering
- K-Means based graph clustering
- Markov clustering
- Minimum Spanning tree based graph clustering

We start from the nearest neighbor approach which is the most simple one,yet most of the times effective enough to solve graph clustering problem.

5.1 K Nearest Neighbor(KNN) Clustering

Nearest neighbor algorithm is an optimization approach for finding closest or most similar observation points/objects where closeness can be expressed using some similarity metric " less similar are the observation objects,the larger is the similarity metric value".

Nearest neighbor graph for n objects (which need to be merged in accordance with some metric) is a graph G with points p and q such that p is the nearest neighbor(NN) of q. p is NN to q since the distance metric from p to q is smallest for any other point from p in graph G [35]. It is a directed graph from p to q since NN relation is not symmetric as the same definition for p to q is not necessary valid for q to p.Fig 5.1 shows the Nearest neighbor graph of 100 observations in Euclidean plane. K-Nearest neighbor is an extension of NN graph

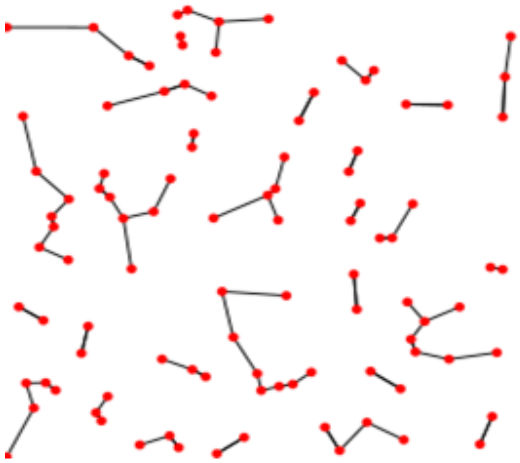

Figure 5.1: NN graph of 100 points in Euclidean plane[35]

methodology in which rather than only combining with the nearest neighbor,the graph node p cluster with the K nearest neighbors.Like NN,KNN relationship is also unsymmetric.
The problem faced while deploying KNN clustering to solve our task is obtaining the correct value of K so that appropriate clustering of crf nodes can be done. Extracting the appropriate value of K is hard in our scenario since even if we fetch some good value of K for a knowledge model,it is not necessary applicable for other knowledge model. Th other model on different Allow node can have different degree of learning. Thus obtaining a global appropriate value of K which is good for every Allow node in the network is unrealistic. Moreover,taking local values of K for each Allow node does not necessary be effective since there is a continuous change in learning across the network.

5.2 K-means based graph clustering

K-means clustering approach basically partition n objects into K clusters where each object become a part of the cluster with the nearest centroids. The K clusters are uniquely represented by their centroids,thus K-means partition the objects into K groups each identified by their centroid [9]. K-means clustering involves two important factors into consideration.

- Distance metric
- Evaluation Criteria

Usually,K means is based on considering Euclidean distance as a distance criteria between the data points.Its evaluation criteria is minimizing the sum of squares of data points from their respective nearest means.

We modified some of the approach used in K means to make it more suitable for accomplishing our task. First,we use Bhattacharya distance as our distance metric for comparing the crf node similarity,it has the similar effect as of Euclidean distance as it also works on the same concept,"more similar objects are,lesser is their distance between them ".
Secondly,since we are basically dealing with the graph objects rather than simple data points in Euclidean space we incorporated the use of adjacency matrix while finding the distance between objects. The adjacency matrix takes into account if there is a connection between the objects being scanned for metric calculation(In our case those objects are crf nodes). We take same approach of minimizing sum of squares of Bhattacharya distance between crf nodes and their corresponding least distanced centroids(again crf nodes which are assigned as centroids).
The majority of methods used in determining the suitable value of K like Gaussian K-means[25],iterative K means are either not appropriate for our task or they didn't provide good results while testing. The Iterative K-means is computationally turned out to be very expensive since range of K can be quite large. In addition to that,clustering is done so many times on single Allow node i.e. both while building,updating local and routing knowledge models. Thus iterative K means (if employed) will be the major hurdle for the scalability of our system.

5.3 Markov Clustering

Markov Clustering is an approach which partitions the graph on the basis of the density of the edges between clusters and within clusters. Generally a graph has a structure where there are many links within cluster and fewer links between the clusters,so MCL extracts the more dense regions of the graph or partial/full cliques (fully connected regions of graph). It is

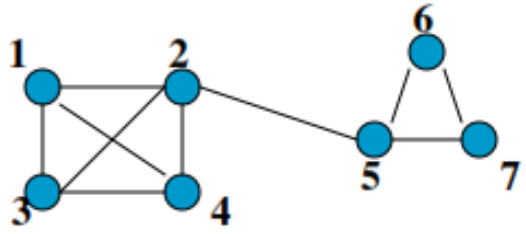

Figure 5.2: Sample Graph for Markov Clustering [38]

conceptually based on Markov Random Walk which can be understandable in a way that if you were to start at any node and randomly travel to its connected nodes,it has high probability that you stay within cluster than to travel across the clusters. Thus,it is based on the flow-based methods like graph cuts based algorithm [32]. It is possible to extract where the

flow tends to gather through random walks and thus finding the near cliques(fully connected subgraph) structures in a graph. These random walks are calculated through Markov chains.
Definition: Markov chain is a sequence of random variables/elements in probability transition matrix where given the present states,past and future states are independent i.e probability on the next time step only depends upon the current probability.
It can be comprehended by a simple example using a graph given in Fig 5.2.At some instance,a random walker present at node 1 has a 33 percent chance of going to node 2,3 and 4,and 0 percent chance to nodes 5,6 or 7.Likewise,transition probability can be produced for other nodes as well to build a transition matrix shown in Fig 5.3.
MCL algorithm basically modifies the random walk process and transition probabilities ma-

	1	2	3	4	5	6	7
1	0	.25	.33	.33	0	0	0
2	.33	0	.33	.33	.33	0	0
3	.33	.25	0	.33	0	0	0
4	.33	.25	.33	0	0	0	0
5	0	.25	0	0	0	.5	.5
6	0	0	0	0	.33	0	.5
7	0	0	0	0	.33	.5	0

Figure 5.3: Transition Matrix for the Sample Graph [38]

trix to emphasize the division of clusters. The edge weights are greater in within cluster-links and lower in between cluster-links during the earlier powers of Markov chains.
The algorithm adjust transition values so that strong neighbors connectivity become more strong and weak neighbors connectivity are demoted in subsequent runs. This adjustment is done through raising a single column of transition Matrix to some positive power(termed as Inflation factor) and then renormalization it. This process is called Inflation. Thus,Inflation process is responsible for strengthen the strong bond further and weakening the already weak associations.This factor controls the granularity of the clusters formed.
During testing,it turned out the approach did not able to cluster properly in case of few similar nodes among the densely connected crf nodes graph. In Fig. 5.4,there are two class of crf nodes according to their similarity of error distributions.The red nodes has almost same distribution so they can be merged together.Similarly,blue nodes have also similar distribution among them.Due to the densely connected underlying crf graph structure,MCL algorithm do not able to distinguish red nodes from the blue nodes and it will consider whole as one cluster (as marked by dotted lines).In this way,it will fail our task of crf graph clustering.
Moreover,finding the right value of inflation factor can also be tricky.It is very much required to do effecting clustering as it decides granularity of clusters like " K " in K-means. The variability of learning across different allow nodes as well as during different instances of time make it really very hard to decide one global good value of inflation factor.

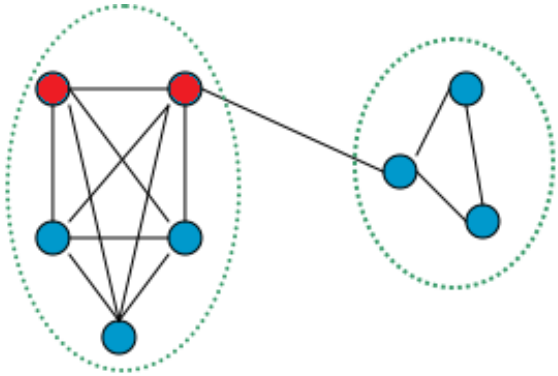

Figure 5.4: Improper Clustering [38]

5.4 Girvan Newman algorithm

Girvan Newman algorithm is a popular algorithm used in community detection in large graphs and it is based on the concept of betweenness centrality. It is a hierarchical method based clustering algorithm used to detect communities in networks [37].

The algorithm works by removing edges progressively from the original graph. It try to

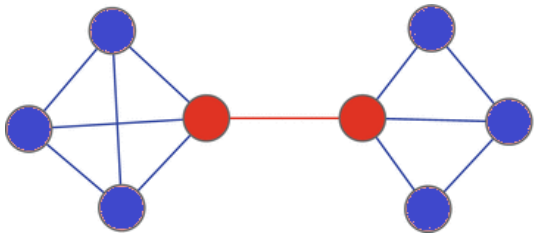

Figure 5.5: Edge Betweenness(mark in red) [38]

extract the most likely edges between the communities,prune them accordingly and thus creating separate communities. GV algorithm produces a dendrogram from top down as graph successively split into different sub graphs by pruning of most likely edges between communities.

The major hurdle faced in incorporating GV algorithm is that it is purely based on the number of edges between the clusters/communities rather than the strength or weight of the edges. Moreover,the algorithm also demands to mention number of edges to be pruned as input parameter. Finding the number of edges to be pruned in our task is really difficult and thus itis unsuitable for our task.

5.5 Minimum Spanning Tree Based Graph Clustering

After exploring different types of algorithms for clustering,we have finally using minimum spanning tree based graph clustering with some modification.The min Spanning tree(MST)

based graph clustering algorithm is indeed a convectional approach and it turned out to be quite suitable and efficient for our task.Now we will dig deeper by understanding the different terminologies in this approach.
In graph theory,spanning tree T is a tree which includes all the vertices of a parent graph.A graph can have many possible spanning tree(shown in Fig.5.6). *Definition* : Spanning Tree is

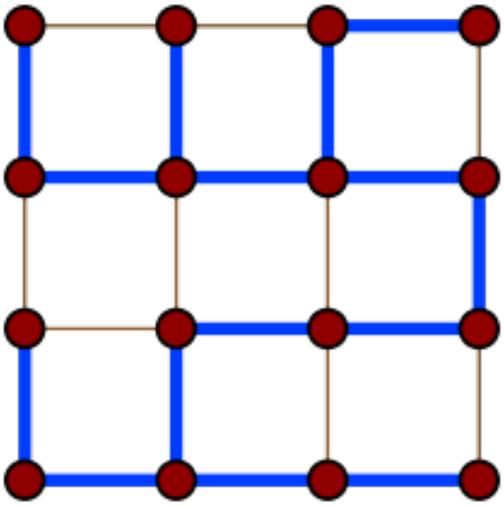

Figure 5.6: A spanning tree (blue edges) of a grid graph [33]

defined as maximal set of edges of a connected Graph G which contains no loops i.e. minimal set of edges which connect all the vertices of graph G [8]. In general edge weights can represent either distance or the similarity between two vertices.In our case,it is the Bhattacharya distance between two crf node's distribution and is an indicator of their similarity of learning in their knowledge models. As stated,there can be more than one spanning tree for a connected graph so graph theory has one more concept of spanning tree with minimum sum of weights.

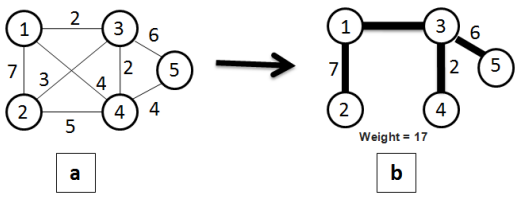

Figure 5.7: Spanning Tree formed from weighted Graph G [24]

Definition :Minimum Spanning Tree for a connected graph G is defined as spanning tree for which the sum of edge weights get minimum,if the edge weights are represented as distance metric.In our case,since Bhattacharya Distance is behaving similar to euclidean distance,convections are applicable in the same way.
It can further be cleared using Fig.5.8,as there are three different spanning trees formed from

the Graph G (Fig.5.8-(a)) but the Minimum Spanning tree is the one with minimum sum of edge weights (Fig.5.8-(b)). There are many algorithms to calculate the minimum spanning

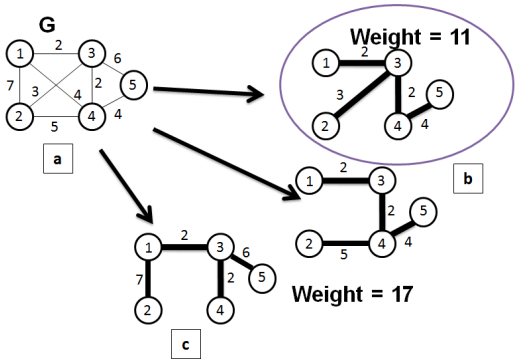

Figure 5.8: Minimum Spanning Tree formed from weighted Graph G [24]

tree(MST) and their improvisations,the widely used are Kruskal's algorithm[28],Prim's algorithm [42] and Boruvka's algorithm[29].
Both are the greedy algorithms to calculate MST.Kruskal's algorithm consider edges in the ascending order of their cost and add them in the result set unless it would create a cycle in the set.Prim algorithm also works in greedy fashion but in different way.It start with an arbitrary vertex of graph and keep growing a tree by adding the cheapest edge (in terms of edge weight) to tree set which has exactly one endpoint in the set T [42]. We have used Prim algorithm in oder to calculate MST as it is using the special data structures (heap) which makes it faster than the other approaches [41].

5.5.1 Prim Algorithm

procedure Prim Algorithm(G: connected weighted graph with n vertices)
Tree := empty minimum-weight edge tree
for *i = 1:n-2* **do**
    e := edge with min weight incident to a vertex in Tree and not forming loop in Tree if added
    Tree := Tree with e added ;
end
return(Tree)

Algorithm 1: Prim Algorithm [42]

The algorithm is also explained pictorially in Fig5.9 for the sample connected weighted graph G.

5.5.2 Convectional MST based algorithm and its improvisation

The convectional MST based algorithm involves the following steps in order to obtain K clusters of vertices from graph G.

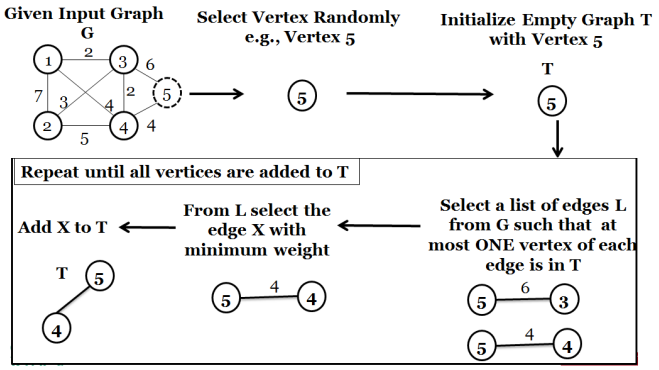

Figure 5.9: Prim Algorithm [24]

- Extract Minimum Spanning Tree from the Graph G.

- Prune K-1 edges from MST.

- Return the resultant K clusters formed.

The result is K non-overlapping collection of vertices of graph but the known challenge which we have faced earlier as well is the determination of the approximate value of K.In order to solve this problem,we are using a penalty based optimization approach to get value of K. The motivation behind this improvisation comes from a thought which manifest maximizing the sum of confidences on each node of the crf graph such that higher weighted edges breaks to form separate clusters. Along with that,there must be some penalty in breaking the edges otherwise the whole graph would break into individual crf nodes. Moreover,these effects are certainly the functions of K. Thus,these are two oppositely behaving phenomenons which have to get balanced at some value of K. The equation provided below manifest the mathematical representation of the above discussed optimization problem.

$$GraphCon_{\text{eff}} = \sum_{i=1}^{i=N} ClusterCon_i - (1 - (\lambda_i / \sum_{i=1}^{i=N} \lambda_i)) * (ClusterCon_i + ClusterCon_{\text{i+1}}) \tag{5.1}$$

The effective graph confidence($GraphCon$) is the sum of confidences of all the clusters at that instance removing the penalty imposed by pruning the specific edge (between $cluster_i$ and $cluster_{\text{i+1}}$).The penalty is the product of the ratios of weights of pruned edge to total sum of edge weights and sum of confidence of new clusters which are created by pruning that edge.

In a nutshell our clustering algorithm works as explained by the below pseudo code.

Available Connected Graph G
extract Minimum Spanning Tree (MST) **from** G **for** *K=1 to K = Number-Edges in MST* **do**
    Prune Edge with Max Edge Weight among all Edges
    Calculate
    $GraphCon_{\text{eff}} = \sum_{i=1}^{i=K} ClusterCon_i - (1-(\lambda_i / \sum_{i=1}^{i=K} \lambda_i)) * (ClusterCon_i + ClusterCon_{\text{i+1}})$
    if $GraphCon_{eff}$ *reaches I Local-Maxima wrt K* **then**
        return resultant K clusters
    end
end

Algorithm 2: Allow MST based Graph Clustering

The minimum spanning tree is extracted from the graph G and starting from K=1 to the number of edges in MST,we check the effective confidence of all clusters with imposing penalty.It is observed that on the value of K where first local maxima is encountered the cluster formed are very pure and fit for our task.We are defining purity in terms of degree of grouping similar clusters. It is quite comprehend able that first local maxima fits this criteria since we are pruning the bad edges which has highest weights(high edge weight is analogous to high degree of dissimilarity),the confidence increases gradually. The penalty imposed on pruning the edges just reverse this effect and there comes a point where pruning actually balance out and the confidence decreases for a while.The effect is temporal and lead to first maxima and then the dissection of this tree into separate crf nodes increases overall confidence faster as more and more vertices are prune apart.It is observed that clusters formed at first local maxima are optimized enough in number as well as in purity to consider this efficient for our task.
The task of clustering crf nodes into clusters of similar degree of learning are achieved by above discussed MST based graph algorithm. Now,in the next chapter,we will discuss the two routing strategies where we can combine our theory developed so far about confidence metric and clustering.

Chapter 6

Knowledge Aggregation based Routing

In this chapter,the routing strategy based on aggregation of knowledge is presented.The previous chapter discussed about the clustering of nodes in a crf graph template residing on every Allow node.As discussed in the system model,each Allow node consists of one instance of local knowledge model called Evoknowldege and one instance of routing model for each of its neighbors.The local knowledge model is actually a set of clusters of crf nodes having similar degree of learning.Theses clusters are called scopes in our terminology.The clustering of crf nodes are done according to the MST based graph clustering algorithm as explained in previous chapter.The routing model of an Allow node for a specific neighbor is an knowledge object which represents the accumulated knowledge of all Allow nodes reachable from that neighbor. Now we will explore more about this routing model and the strategy developed to create this routing model using aggregated knwoldege propagation.

6.1 Routing Protocol Overview

The knowledge models can contain huge number of crf nodes so it is bandwidth inefficient to forward confidence score of each node of the model in the whole network. As explained in previous chapter,we aggregated the confidence scores of set of crf nodes into clusters. Now,these cluster information need to be propagated to other Allow nodes in the network.The aggregated information or the combined confidence information of the scopes are propagated on remote nodes to create/update the routing model.
Definition: Routing Model is the knowledge model created at local Allow node from combining the summarized knowledge about remote models received from neighboring Allow nodes with the local knowledge model.
Since the template of knowledge model is same on each Allow node,when the clusters information is received,they are combined to create a routing knowledge model. In figure 6.1,it is depicted as clusters information of the models of neighboring nodes are combined along with the local model to create an effective routing model at the local Allow node.As it is explained already,normal distributed CI information is taken and mean and deviation of error will be preserved for each scope as CI measure.
Steps involved in the routing algorithm : Pre-Routing Step :Clustering of crf node (based

on MST graph clustering) are done to form scopes. The resultant clusters have aggregated error distributions from individual crf nodes by combining confidence Interval of the individual crf nodes.Pooling of Confidence Intervals is done according to section 4.2 and equivalent mean and margin is considered for each crf node in a scope(cluster) for any further computation.The confidence value of each crf node is the value obtained combining equivalent mean and margin.

1 Formation/Update of routing model through the combination of scopes from local knowledge model and routing information of neighboring Allow nodes. The routing information for each neighbor is stored in routing table on local node. The routing table entries contains neighboring Allow Node Id and the list of scope Information.
Creation of routing model further involves two important steps.

- Combining the error distribution on each crf node obtained from different scopes obtained from local model and from neighboring nodes. Combining involves taking the best scope information for each crf node i.e. the scope information(mean,margin and number of observation points) with maximum confidence value are considered as resultant confidence interval information for each crf node while merging.

- Clustering of crf nodes with new and combined confidence interval information. It will create new scopes which form the routing model.

2 The routing model formed for each neighboring nodes contains the list of newly formed scopes. As stated,Each scope is a tuple having Entity Id,confidence Interval Information and list of crf nodes in the scope.

3 Each routing model is then propagated to its respective neighbors.The propagation involves the forwarding of list of scopes information specific to routing model.

4 The neighboring Allow nodes update their routing table entry when they receive information from this Allow Node. Basically,it involves receiving the new scope information and merging the received information with the already present scopes. Then routing model update is done which involves steps 1 from above.

This is a superficial overview of the routing model creation,details are provided in the algorithm section afterwards. Fig 6.2 shows that on the local Allow node,how means and deviations are stored to represent confidence measure both for local and routing knowledge model.Each routing table entry has neighboring Allow node Id and list of scopes.

6.2 Routing Algorithms

In this section,we will talk about the algorithms used for routing the test query to the destined Allow node along with the mechanism of how the forwarding of summarized knowledge takes place in the network.

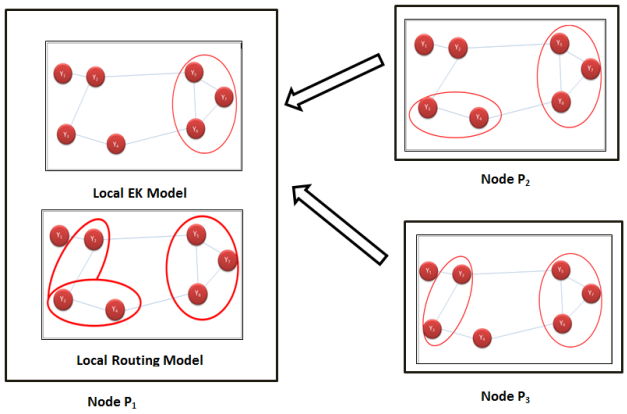

Figure 6.1: Formation of Routing Model

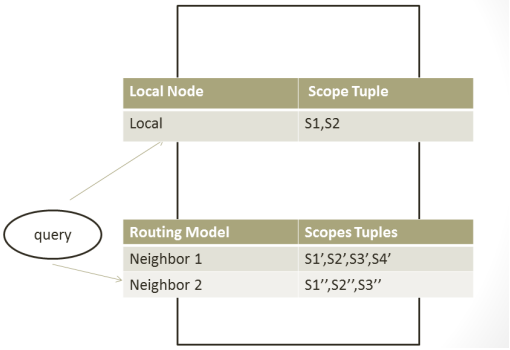

Figure 6.2: Routing Table

6.2.1 Generic Data Structures

We first need to highlight some of the following data structure used in general.

- Confidence-Information
  Mean
  Margin
  Instances
  Confidence
  The data structure which represents the distribution information (mean,deviation and number of data points(instance)) and its associated confidence value.

- Scope-Information
  List : Crf Nodes in Scope
  Confidence-Information
  Scopes are the collection of crf nodes clustered together to represent aggregated knwoldege about its member crf nodes.

- Knowledge Model : List<Scope-Information>
  The Knowledge model object represents the aggregated information of the local knowledge model. It provides the local confidence which can be delivered by that Allow Node specific to test query.It is basically a list of Scopes which represent each cluster information in the model.
- Routing Model: List<Scope Information>
  Routing Model imitates the local knowledge model in anatomy but while there is single knowledge model for each Allow node,Routing Model object is one for each neighbor specific to one Allow Node.
- crf Graph <CrfNodeId,Confidence-Information>
  Represents the error distribution information associated for each of the crf node identified by crf Node Id.
- Routing Table: Map<Neighbor,Routing Model>
  Routing Table which contains a Map of Routing models one for each neighbor.
- Allow Node
  List of neighboring nodes
  CRF Graph containing <CrfNodeId,Confidence-Information>
  Knowledge Model
  Routing Table
  Allow node data structure is modeled to represent Allow Node Entity. It contains list of neighboring nodes,crf map and knowledge model object.

6.2.2 Routing Protocol Sceneries

Now we will look into the logics implemented at various scenarios of routing strategy for knowledge aggregation. There are following sub parts for our routing strategy.

- Routing test query to destination
- Node receives response message
- Creation of Routing Model
- Aggregated Knowledge Forwarding

Routing test query to destination

When a node P_i receive query from P_j,first it will check whether its local knowledge model can answer the query with the specified confidence mentioned. If the local knowledge model has higher confidence than specified in query,it will answer else it will check its routing table to obtain the best confidence it can get from its neighbors to forward the query. It will forward the query to that neighbor whose routing model will give highest confidence specific to the query.

As mentioned Routing model for each model imitates local knowledge model but it is a representation of the associated neighbor knowledge and it will answer the degree by which any query can be answered by that neighbor. Thus,for any query,if it is not answered by the local knowledge model,which ever routing model deliver highest confidence,the query propagation will be done to that neighbor.

Available Knowledge Model(KM),Table of Routing Models(RT)
on Receive (Query q= $(q, C_{\min})$) **from** node P_j **do:**
if $(C_{\min} < C_{\mathrm{i}}(q))$ **then**
send(Response,$C_{\mathrm{i}}(q)$) **to** $nodeP_j$
else
send(Query q= $(q, C_{\min})$) **to** bestNeighbor(q)
Algorithm 3: Node receives query message

Node receives response message

The allow node sends the response associated with the query if it can satisfy query's requirement along the same path query traveled. If the node received the response finds its updated confidence is higher than confidences received in response message,it will replace it with new confidence and sends response message further back to the same path.

Available Knowledge Model(KM),Table of Routing Models(RT
on Send (Receive Message m = (q, C_{current})) **to** node P_j **do:**
if $(C_{\text{current}} < C_{\mathrm{i}}(q))$ **then**
send(Response,$C_{\mathrm{i}}(q)$) **to** $nodeP_j$
else
send(Response,C_{current}) **to** $nodeP_j$
Algorithm 4: Node receives response message

Creation of Routing Model

On node P_i,in order to create a routing model for neighbor P_j,the local knowledge model and all the routing model from the routing table except model of P_j are merged together. The merging involves combining the scope information from all models. For each of the crf node,Confidence-Information structure with highest confidence will get associated from all the scopes of all models which contains that particular crf node. Thus,the routing model formed has each of its crf node having best confidence information. In this way,the goodness of the learning in crf nodes can be preserved and propagated to next level.
Afterwards,re-clustering is done on the crf node graph which has updated CI information on each of its crf nodes. The re-clustering using spanning tree based clustering algorithm

produces new scopes and sets of new formed scope-information is then propagated to respective neighbors.

Available Knowledge Model(KM),Table of Routing Models(RT),Routing Models(RM),crfGraph<crfNode,Conf-Information>
on node P_i CreateRM **for** node P_j **do:**
fetch while *crfNode* $\in$ *crfGraph* **do**
  while *neighbors except* P_j **do**
    fetch Conf-Information $\hat{=}$ $Confidence_{\max}$
    $Confidence_{\max}$ = max(Confidence $\forall RM_{\text{neighbors}}$)
    populate RM(Conf-Information)
  end
  do scopes = Clustering(RM) **return** scopes
end

Algorithm 5: Knowledge Aggregation based Routing Model Creation

Aggregated Knowledge Forwarding

When node P_i has some update in its local knowledge,it will send its updated knowledge to its neighbors. First,it will call clustering on its knowledge model and create routing model for each of its neighbors. The creation of routing model is done according to the algorithm described previously. The new scopes are formed in each routing model and accordingly updated messages are sent to each neighbor.

Available Knowledge Model,Table of Routing Models
procedure Forwarding() **from** node P_i **to** node P_j **do:**
on node P_i scopes = CreateRM **for** node P_j send(scopes) **to** $nodeP_j$

Algorithm 6: Knowledge Forwarding

We have defined and analyzed the new routing strategy based on knowledge aggregation and provided algorithms related to various sceneries. The major problem with this approach is message overhead increases as the network expands since updates messages need to be propagated for every small change. Thus,it hinders the scalability of our system with very large number of both Allow and crf nodes. Now we will discuss the other approach in next chapter which will mitigate this problem.

Chapter 7

Query Learning Based Routing Protocol

The knowledge aggregation approach as discussed previously limits the scalability of system to an extent. In a very large and dynamic networks,the degree of message overhead increases sharply since update of local knowledge model need to be propagated in the network. The continuous updates make the task of maintaining the routing tables difficult and very expensive. In order to mitigate these drawbacks,we have come up with a new approach of learning the patterns in queries across the network rather than maintaining the routing tables comprising aggregated neighbor knowledge information. Thus,Query Learning approach is a new strategy which develop routing by learning the behavior of the past queries.
We have first provided the QL based routing protocol overview,followed by its detailed description and functioning. The description explores the machine learning approach used in the learning and it finally summarized with suggesting the additional learning methods that can be employed to enhance the query learning process.

7.1 Protocol Overview

The section gives the superficial idea of this approach. We are maintaining a single instance of local knowledge model and one instance of routing model for each neighbor at an Allow node. Further,we will discuss how a routing model is created and its functioning. We will then discuss the propagation of test query for this approach.

7.2 Knowledge Model and Confidence Estimation

In this section,we will explore the confidence estimation and representation of the local knowledge model.
The knowledge model creation on the Allow node is done in the same way as that in Knowledge aggregation approach. Allow node has a template of crf graph and each crf node is associated with error distribution which gives the degree of learning and its associated reliability. The creation of knowledge model is done grouping the similar nodes on the basis of their residing distribution and between edge probability. The clusters (so called scopes in our

terminology) are formed using our previously elaborated minimum spanning tree based graph clustering approach. The previous chapters can be revisited to get the better understanding of the developed concept.
The confidence metric is same like in previous approach.It can be extracted using the parameters of the error distribution residing on each crf node.
The idea of query learning can simply starts by creating a routing model corresponding to each neighbor. The routing model is build using regression method in which the training data is created from the special type of queries sent across the network through the process of Exploration. The test queries will then act as a test data for this underlying regression of routing model to deliver the results.
In order to populate the routing model with this information,each Allow node trigger the inspection queries to gather information about its neighbor. These inspection queries are called Exploration Query and these are issued in timely manner. Thus,there are two types of queries in this approach propagating across the network.

- Exploration Query It is used to inspect and gather the information about the neighbor. It is issued in timely interval to its neighbor and collects the learning information along that path.

- Test Query It is a normal query which is same like what we have in knowledge aggregation approach. It is triggered on any Allow node to destined for the best node in the network which can satisfy its minimum confidence expected.

Now we will explain Exploration process in detail.

7.2.1 Exploration

It is the process of inspecting the network by issuing the exploration queries to the neighbors and extracts the feedback from their respective paths. The feedback corresponding to the explored query populates the routing model of its respective neighbor. The extent of Exploration applied on the network depends upon certain parameters.

- Selectivity : This parameter specifies the number of neighbors for which query need to be forwarded. Let's say when the selectivity is three,query message is forwarded to three randomly chosen neighbors and similarly at every hop same decision is taken while forwarding the query.

- Hop Count: It specifies the extent of exploration by reaching to the provided number of Allow nodes. The exploration query is forwarded with maximum number of hop counts.

Let's say when we mention s=3,h=3,at every allow node,query is forwarded to 3 randomly chosen neighbors till in total for each path maximum hop count is reached (shown in Fig.7.1).

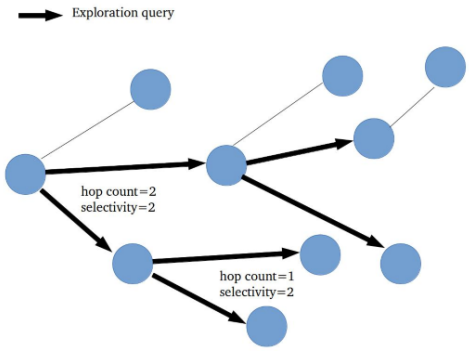

Figure 7.1: Selectivity [13]

Exploration Query

First we need to understand the anatomy of the exploration query to further discuss the query learning approach of routing. Exploration query comprises of the following members :

- Global observed parameters: Eg Weather and Time of day
- Bitmap for crf nodes: Each bit represents the crf node in the common template. Bits are set for the nodes for which queried is made. In Fig7.2. B for 6 nodes crf graph,if query is made for 1,2 and 3 crf node bitmap would be 111000.
- Max Confidence: This field store the maximum value of the feedback as the query object propagates along the path. The feedback in our case is the confidence on every hopped allow node. The max confidence is stored and consider as the final value which will be returned as an feedback at the source node.

Fig 7.2 shows the sample of exploration query created with the given crf graph,it contains the global observed parameters and bit map of crf nodes queried and max confidence field which will be populated while hopping across the network.

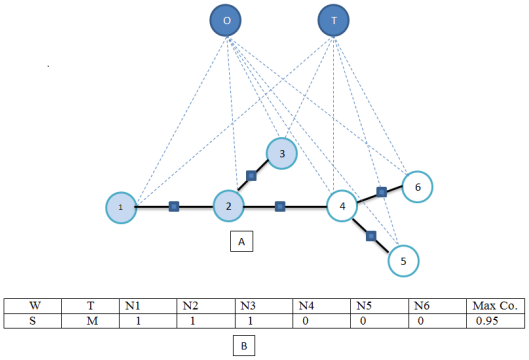

Figure 7.2: ExplorationQuery

The algorithm of Exploration is briefly described below. While the selectivity and hop count remains below their maximum values,the exploration query triggered at P_i continue to hop neighbors and extract the maximum value of confidence in its path.

Available Knowledge Model,Table of Routing Models
on trigger Exploration (Exploration Query q(q,C_max),selectivityMax,hopCountMax)
from node P_i
while *(selectivity < selectivityMax and hopCount < hopCountMax)* **do**
  if ($C_{\max} < C_{\mathrm{KM(q)}}$) **then**
  $C_{\max} = Confidence_{\mathrm{KM(q)}}$
  selectivity =selectivity +1
  hopCount = hopCount +1
  send(Query q= (q, C_{this})) **to** RandomNeighbors(q,selectivity,hopCount)
end
send($C_{\max}$) **to** $nodeP_i$

Algorithm 7: Exploration

Each Allow node gives its confidence for the scope of crf nodes specified in the query. This confidence value is the feedback provided to the source Allow node wrt query. The query object keep the maximum of the confidence value it come across while hopping and return it back to the source Allow node. This approach can be considered as analogues to immediate feedback mechanism since it take the local information of the node while hopping. We can expand it to the delayed feedback mechanism in the future work.
Allow nodes trigger Exploration process which we have discussed in last section to update its routing models. The query and its maximum confidence is retrieved and will use as an information to build routing model.

7.3 Routing model Creation

The routing model of a neighbor node is an object which represents the learned information about the nodes reachable by that neighbor. In order to populate the routing model with this information,each Allow node trigger Exploration to gather information about its neighbor.
The approach differs in starting point of the query,in a way that test query can ideally be triggered only from that Allow node on which exploration queries were issued. It is quite understandable since Exploration build and updates the routing model. When there is no Exploration done on any Allow node,its routing models for the corresponding neighbors are empty.
Thus,it depends upon the user how he wants a system to behave for the test query. If the requirement of the system is such that test query should be triggered on any of the node,then exploration must be done on each node of the network. On the other hand,if the proposed system behavior favors only few testing points (from which test querying is done),then exploration can be done only on those nodes. Thus,this behavior also leads to the scalability and adaptation for very large and dynamic networks when changing the above

mentioned system requirement.

W	T	N1	N2	N3	N4	N5	N6	Max Co.
S	M	1	1	1	0	0	0	0.95
S	T	1	0	0	1	1	1	0.98
S	M	1	1	1	0	0	0	0.95
W	T	0	1	1	1	0	0	0.87
W	M	1	1	1	0	0	0	0.17
W	T	1	1	0	1	1	0	0.23
A	M	1	1	1	0	0	0	0.95
A	T	0	0	0	0	1	1	0.35
S	M	1	1	1	0	0	0	0.92
W	T	1	0	0	0	1	1	0.11
W	M	1	1	1	0	0	0	0.85

Figure 7.3: Training Data for Regression of Routing Model

As discussed,Allow nodes collect the maximum confidence along the path of its neighbors by triggering Exploration process in timely manner. This information which arrived through a single exploration query act act as a single instance of information to build the routing model. In order to build the routing model,we need several such instances of information (q,Cmax),so several exploration queries are issued in one cycle of Exploration process. These pairs of (q,Cmax) collected acts as training data to learn the routing model for each neighbor (Fig 7.3). These pairs are different for different neighbors and thus the learned parameters of routing model obtained from them. We are using standard machine learning approach of regression to build routing model and its training data can be considered by q as input and Cmax as output. The algorithm given about the Routing model creation explains the procedure of creation.
Available Training Data<Pair(Q,$Confidence_{\max}$)>
procedure RoutingModel **on** node P_i **do:**
Matrix X = Training Data(Q)
Matrix Y = Training Data($Confidence_{\max}$)
$Matrix\ \lambda$ = Regression(X,Y)
return λ

Algorithm 8: Query Learning Based Routing Model Creation

The routing model is created using regression from the training data obtained from exploration. The routing model is actually stored as a set of regression coefficients(λ).Fig 7.4 shows the detailed view of the routing model formed at Allow node.

7.4 Propagation of Test Query

When the test query arrives at any node which has the routing models created using exploration process,it will be checked whether the query can be answered locally. If the local model can not able to satisfy the query's confidence requirement,confidence of each routing model is calculated by multiplying query parameters with regression coefficients provided in the algorithm given below. The query is then propagated to the best neighbor which corresponds to

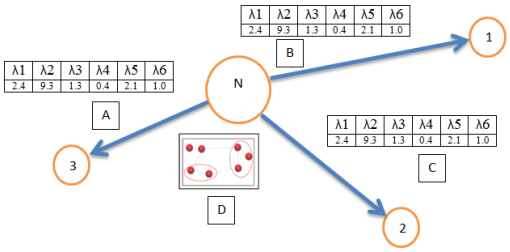

Figure 7.4: QL based Routing

the one with maximum confidence.

Available Knowledge Model,Table of Routing Models
on Receive (Query q = $(q, Confidence_{\min})$)
from node P_j **do:**
if $(Confidence_{\min} < Confidence_{\mathrm{KM}}(q))$ **then**
send(Response,$Confidence_{\mathrm{KM}}(q)$) **to** $nodeP_j$
else
send(Query q= $(q, Confidence_{\min})$) **to** BestNeighbor(q)

Algorithm 9: Node receives test query message

Extracting the best neighbor to which query need to be propagated is given in the below algorithm.

Available Table of Routing Models(RT)
procedure Best Neighbor(q) **on** node P_i **do:**
RM_i = getRM from RT
$Confidence_i = RM_{\mathrm{i}}(q)$
MaxConfidence= max($Confidence_i$)
BestNeighbor = $(Neighbor_i \mathrel{\widehat{=}}$ MaxConfidence
return BestNeighbor

Algorithm 10: Best Neighbor(q)

Multiplication of query parameters with regression coefficients to get confidence of routing model is provided.

Available Matrix λ
procedure RM(Query q(Q,$C_{\min}$)) **on** node P_i **do:**
Confidence = $\lambda * Q$
return Confidence

Algorithm 11: Query Handling by Routing Model

Thus,we have explored the approach of learning in which the inspecting queries are used to build a system which have efficient routing capabilities for any new query.Moreover,the message overhead for propagating the aggregated information of knwoldege model is mitigated.

Now we will discuss the results we obtain by testing the two routing strategies.

Chapter 8

Evaluation

The chapter deals with the evaluation of the two routing strategies we have explored so far. We evaluate Knowledge aggregation and query learning routing strategy w.r.t. to the accuracy of the retrieved confidence values. The retrieval information quantifies the degree of effectiveness of our concepts employed in two strategies.

Definition :Accuracy of the system deploying a specific routing strategy is the degree by which it can retrieve the high confidence value for the query.

The retrieval of high confidence values means extracting the better learning regions across the network. Ideally,the system should always reach the highest region of learning wrt to query. The effectiveness of the routing strategy make sure that the specific query should reach the node with best possible learning region. Thus,accuracy is defined as the ratio of the confidence that was retrieved to the globally best confidence value in the knowledge model of any node in the system. In the subsequent sections,we will discuss the topology used for evaluation,crf models used and finally we will evaluate the accuracy of the strategies for different scenarios using different parameters configurations. We are using Peersim Simulator[34] to implement our two routing strategies.

8.1 Topology

We are using the tree based acyclic topologies as to avoid the additional complexities associated with the loops in the graph.The topology generator module is implemented which generates the tree based structure starting from ten allow nodes to five thousands. Each allow node has a probabilistic graph i.e crf graph residing which represents the container of its knowledge. The crf graph is again generated by topology generator but the main difference is that it is not that large and dense.It is imitating the general crf graphs used in various real life applications. The template of crf graph is same on each allow node only the degree of learning can differ,through their different error distributions on each crf node. The example of the crf graph used as a common template in provide in Fig 8.1.

In the next section we will evaluate the knowledge aggregation based routing strategy.

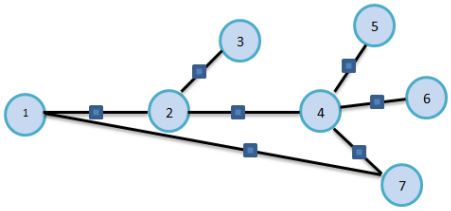

Figure 8.1: Simplest CRF graph Used

8.2 Knowledge Aggregation based Routing

This routing strategy is evaluated keeping in mind the dynamics and scalability of the system. In order to take into account the system is tested from 10 to 5000 Allow nodes.We are evaluating in a scenario keeping the number of allow nodes fixed and sending the test queries initialized on random nodes,we trace the behavior when updates are send increasingly to more neighbors. The number of neighbors for which updates are sent increases one to all neighbors and then change in accuracy is observed by sending test queries from random Allow nodes. As you can see as the sending of updates are increasing to more available neighbors at each allow node,accuracy of the system is increase gradually.

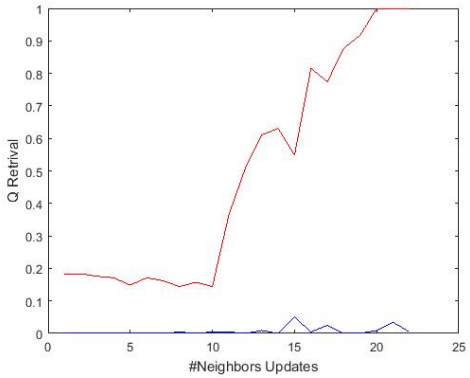

Figure 8.2: Accuracy vs Updates sent to Number of Neighbors

In Fig. 8.2,plot is drawn as Accuracy of Retrieval versus the Updates sent to number of neighbors. There are 4000 Allow Nodes in the network having a crf graph of 20 nodes on each of them. Each Allow node is connected to maximum of 20 neighbors. As we can see in the plot,the rate of change in accuracy is quite low and is almost constant till when the amount of updates are sent to nearly half of the neighbors at each Allow node. The second half see a steep rise in the accuracy value and as the more and more neighbors get updated accuracy value increases. It reaches the maximum value just before when the updates get propagated to all the neighbors at each Allow node. We can compare accuracy of our knowledge aggregation

at any instant with the random flooding (marked in blue) which is very low and does not change with increase much in neighbor updates.

Similarly,we can test the extent and scalability of our routing scheme by increasing the crf nodes in the template graph at each Allow node and then see its effects.Fig 8.3 shows the plot with 1000 Allows nodes having 1000 crf nodes in their graph. The gradual increase in the accuracy can be seen in to contrast to the previous plot with high number of Allow nodes and low crf nodes.As we can see even with 1000 Allow node and 100 crf nodes , it is able to achieve 75 percent accuracy when updates are sent to 15 neighbors. Thus, it is able to achive 75 percent accuracy when 75 percent of total neighbors are been updated.

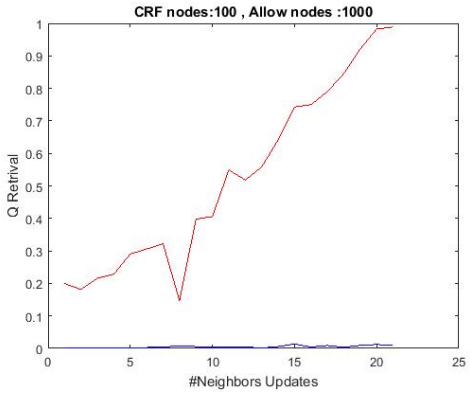

Figure 8.3: Accuracy vs Updates sent to Number of Neighbors

8.3 Query Learning based Routing

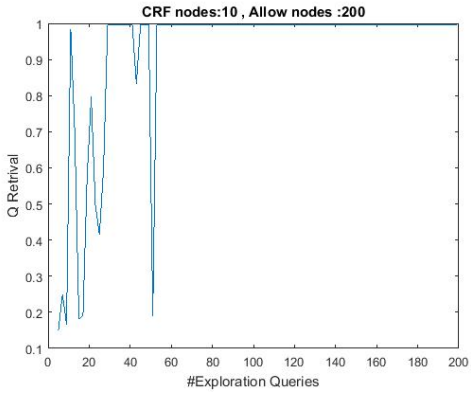

Figure 8.4: Query Learning: Accuracy vs Exploration Queries

As discussed in the previous chapter,the learning of routing models are done using regression and the training data required for this regression process is collected through Exploration.

The exploration queries are sent which gather the maximum confidence specific its query and other parameters like selectivity and hop count.

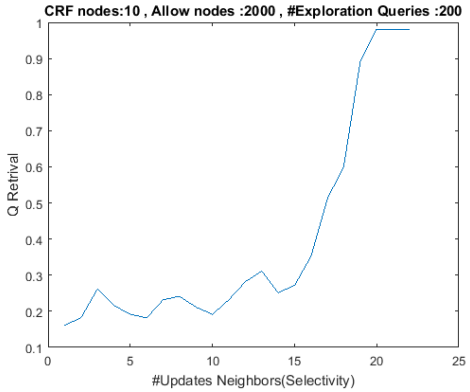

Figure 8.5: Query Learning: Accuracy vs Sensitivity

Fig 8.4 shows the plot of accuracy drawn against the number of exploration queries. As we can see,the plot shows the abrupt behavior till reaching to the number of exploration queries which are enough to build a routing model using regression analysis. Once regression based routing model got enough training data (exploration queries),the accuracy got saturated to its maximum.

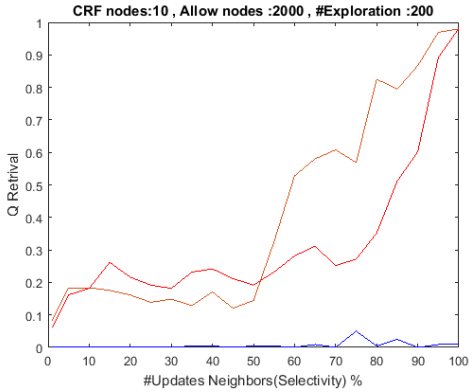

Figure 8.6: Accuracy vs Updates sent to Number of Neighbors
Knowledge Aggregation(Orange), Query Learning(Red),Random(Blue)

Selectivity defines the number of neighbors to which exploration is need to be done while Hop Count tells the depth of the exploration. The parameter settings can directly affects the retrieval behavior and thus accuracy of the system.

Fig 8.5 draws the accuracy against the selectivity as defined previously. The number of neighbors are maximum of 20 for each Allow node and we increases selectivity up to maximum

available neighbors at each node. The plot reveals that unless selectivity is quite high,retrieval accuracy remains low.

Fig 8.6 draws the accuracy against the selectivity as defined previously. The number of neighbors for which updates are sent are expressed in percentage.As we can see , in knowledge aggregation, the increase of accuracy is gradual after 50 percent of neighbors are updated while in query learning most of the neighbors need to be updated to achieve similar results.If we want to achieve a decent accuracy of around 70 percent , we can achieve it with less selectivity through Knowledge aggregation routing scheme.

Chapter 9

Conclusion and Future Work

This thesis has the major task of designing and implementing a system which supports efficient knowledge retrieval mechanism of probabilistic models stored on distributed nodes. The system uses undirected probabilistic model called conditional random fields to model the underlying observed data to build the learning system. The essence involves developing a routing strategy which can route any query to the highest degree of learn model in the distributed network. The concept of confidence based retrieval is exploited and confidence metric is developed which can quantify the degree of learning on CRF model residing on every node in the network. The query must be answered by the system using this metric for efficient routing to best learned model node.
The major challenge was to quantize the quality of learning in CRF knowledge model as well the reliability involves in that learning in absolute sense. We use the statistical concept of confidence interval and combines parameters of both accuracy and reliability in a single quantity to get this saturated sense of learning at every model. Moreover,routing to the node which has best learned CRF model in a heterogeneous and dynamic environment poses an additional challenge. We have developed two different approaches of routing to tackle and maintain scalability in these complex scenarios. The clustering approach we used for summering the knowledge provides a distinct way to handle graph clustering. There are certain scenarios where there are tradeoffs like message overhead in the network versus clustering in aggregation based routing,penalizing graph clustering technique and selectivity and hop count selection in query learning routing strategy.
In the evaluation,we have extracting the quality of the system by its retrieval accuracy i.e. what confidence is retrieved to the global best specific to the query parameters. We are employing tree based topology,graphs poses an additional complexity and can be further investigated in future work. We have evaluated the knowledge aggregation technique up to 5000 nodes in a network and 500 crf nodes in a graph to test its scalability. Similarly,we have tested query learning with changing sensibility and with increasing number of exploration queries to test the limit of its scalability.
The increase in the number of nodes in crf graphs drastically effect the accuracy of the QL routing approach. In order to mitigate this problem,dimensionality reduction /feature extraction must be applied on regression used in building query learning based routing models. Likewise,message overhead can further be reduced by investigating the approach of propagating summarized information only for network nodes where confidence saturation has not been reached. The major technical contribution of this work involves developing routing strategies to summarize probabilistic knowledge models based on efficient graph clustering and merging

aspect of learning accuracy with its reliability. The merging of accuracy and reliability of learning model through combining mean and deviation of error possess various philosophic alternatives. The confidence metric as discussed is not final and can be devised according to the user attitude of risk handling while deciding the distributed system behavior. It further takes a dig into game theory which can further be pursued as a future direction of this research.
On the ending note,we can assert that our approach of summarizing knowledge in a probabilistic model and propagating it and finally its effecting retrieval can pose great direction to further enhance and develop interesting applications that can make good use of our research work.

Bibliography

[1] *J. Gubbi,R. Buyya,S. Marusic,M. Palaniswami. Internet of Things (IoT): A vision,architectural elements,and future directions. Future Generation Computer Systems,29(7):1645–1660,2013.*

[2] *A. Jacobs. The Pathologies of Big Data. Commun. ACM,52(8):36–44,2009. doi:10.1145/1536616.1536632. URL http://doi.acm.org/10.1145/1536616.1536632.*

[3] Stuart *Cox D.R.,Hinkley D.V. (1974) Theoretical Statistics,Chapman Hall, a b Kendall,M.G. and Stuart,D.G. (1973) The Advanced Theory of Statistics. Vol 2: Inference and Relationship,Griffin,London. Section 20.4*

[4] *http://www.cmswire.com/cms/featured-articles/how-semantic-web-tech-can-make-big-data-smarter-026726.php*

[5] Yale University *http://www.stat.yale.edu/Courses/1997-98/101/confint.htm*

[6] Bhattacharyya *Bhattacharyya,A. (1943). "On a measure of divergence between two statistical populations defined by their probability distributions". Bulletin of the Calcutta Mathematical Society 35: 99–109. MR 0010358.*

[7] *An Introduction to Conditional Random Fields for Relational Learning,Charles Sutton,University of Massachusetts,USA*

[8] Graham *R. L. Graham and Pavol Hell. "On the History of the Minimum Spanning Tree Problem". 1985.*

[9] MacQueen *MacQueen,J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability 1. University of California Press. pp. 281–297. MR 0214227. Zbl 0214.46201. Retrieved 2009-04-07.*

[10] *R. Bryant,R. H. Katz,E. D. Lazowska. Big-Data Computing: Creating Revolutionary Breakthroughs in Commerce,Science and Society,2008.*

[11] *R. Blumberg,S. Atre. The problem with unstructured data. DM REVIEW,13:42– 49,2003*

[12] Richard *David Richard Schäfer,Santiago Gómez Sáez,Thomas Bach,Vasilios Andrikopoulos,Muhammad Adnan Tariq: Towards Ensuring High Availability in Collective Adaptive Systems. In Proceedings of the International Workshop on Business Processes in Collective Adaptive Systems (BPCAS 2014).*

[13] Thomas *Bach,Thomas; Tariq,Muhammad Adnan; Mayer,Christian; Kurt Rothermel: Utilizing the Hive Mind - How to Manage Knowledge in Fully Distributed Environments. In: OTM 2015 Conferences. Universität Stuttgart,Fakultät Informatik,Elektrotechnik und Informationstechnik.*

[14] Tariq *Muhammad Adnan Tariq,Boris Koldehofe,Gerald Koch,and Kurt Rothermel,Distributed spectral cluster management: a method for building dynamic publish/-subscribe systems,Proceedings of the 6th ACM International Conference on Distributed Event-Based Systems Pages 213-224*

[15] *X. Wu,X. Zhu,G.-Q. Wu,W. Ding. Data mining with big data. Knowledge and Data Engineering,IEEE Transactions on,26(1):97–107,2014*

[16] *S. Lohr. The age of big data. New York Times,11,2012*

[17] *D. Agrawal,S. Das,A. El Abbadi. Big Data and Cloud Computing: New Wine or Just New Bottles? Proc. VLDB Endow.,3(1-2):1647–1648,2010. doi:10.14778/ 1920841.1921063. URL http://dx.doi.org/10.14778/1920841.1921063.*

[18] *MapReduce: A Flexible Data Processing Tool Jeffrey Dean,Sanjay Ghemawat Communications of the ACM,Vol. 53 No. 1,Pages 72-77 10.1145/1629175.1629198*

[19] John W. Dower *Readings compiled for History 21.479.* 1991.

[20] L.R. Rabiner *A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE,77(2):257286,February* 1989.

[21] Tang *C. Tang,Z. Xu,and M. Mahalingam. pSearch: Information retrieval in structured overlays. In First Workshop on Hot Topics in Networks (HotNets-I),Princeton,NJ,2002.*

[22] Nguyen*F. M. Cuenca-Acuna and T. D. Nguyen. Text-Based Content Search and Retrieval in ad hoc P2P Communities. Technical Report DCS-TR-483,Department of Computer Science,Rutgers University,2002.*

[23] Andrew McCallum *Confidence Estimation for Information Extraction*

[24] $http : //www.csc.ncsu.edu/faculty/samatova/practical - graph - mining - with - R/slides/ppt/Graph_Cluster_Analysis.pptx$

[25] $http : //www.cse.psu.edu/CSE586Spring2010/lectures/cse586gmmemPart1_6pp.pdf$

[26] Carzaniga *Architectures for an Event Notification Service Scalable to Wide-area Networks. PhD thesis,Politecnico di Milano,Milano,Italy,Dec. 1998.*

[27] Amir *Oertel,P. and Amir,E. A framework for commonsence knowledge retrieval,Proceedings of the 7th International Symposium on Logic Formalizations ofCommonsense Reasoning,2005.*

[28] Kruskal *Kruskal,J. B. (1956). "On the shortest spanning subtree of a graph and the traveling salesman problem". Proceedings of the American Mathematical Society 7: 48–50. doi:10.1090/S0002-9939-1956-0078686-7. JSTOR 2033241.*

[29] Borůvka *Nešetřil,Jaroslav; Milková,Eva; Nešetřilová,Helena (2001). "Otakar Borůvka on minimum spanning tree problem: translation of both the 1926 papers,comments,history". Discrete Mathematics 233 (1–3): 3–36. doi:10.1016/S0012-365X(00)00224-7. MR 1825599.*

[30] Guy B. Coleman,Harry C. Andrews *Image Segmentation by Clustering,Proc IEEE,Vol. 67,No. 5,pp. 773–785,* 1979.

[31] Hald *Hald,A. (1998),A History of Mathematical Statistics from 1750 to 1930,John Wiley and Sons,ISBN 0-471-17912-4.*

[32] *Stijn van Dongen,Graph Clustering by Flow Simulation. PhD thesis,University of Utrecht,May 2000*

[33] $https://en.wikipedia.org/wiki/Spanning_tree/media/File:4x4_grid_spanning_tree.svg$

[34] $http://peersim.sourceforge.net/$

[35] Altman,N. S. *"An introduction to kernel and nearest-neighbor nonparametric regression". The American Statistician 46 (3): 175–185. doi:10.1080/00031305.1992.10475879* 1992.

[36] Bhattacharyya,A. *On a measure of divergence between two statistical populations defined by their probability distributions,Bulletin of the Calcutta Mathematical Society 35: 99–109. MR 0010358*

[37] Girvan M. and Newman M. E. J.,*Community structure in social and biological networks,Proc. Natl. Acad. Sci. USA 99,7821–7826* 2002.

[38] $https://www.cs.ucsb.edu/xyan/classes/CS595D-2009winter/MCL_Presentation2.pdf$

[39] Andrew McCallum *Motion Clustering and Estimation with Conditional Random Field,Proceedings of the IEEE,*

[40] Thomas G. Dietterich.*Machine learning for sequential data: A review. In Lecture Notes in Computer Science. Springer Verlag,* 2002.

[41] MST,`https://www.ics.uci.edu/~eppstein/161/960206.html`

[42] Princetion MST, `https://www.cs.princeton.edu/~rs/AlgsDS07/14MST.pdf`

[43] Charles Sutton *An Introduction to Conditional Random Fields for Relational Learning*

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

place, date, signature