Sensor-Based Understanding of Daily Life via Large-Scale Use of Common Sense

William Pentney, Ana-Maria Popescu,
Shiaokai Wang, Henry Kautz
Department of Computer Science & Engineering
University of Washington
Box 352350
Seattle, WA 98195
{bill,amp,shiaokai,kautz}@cs.washington.edu

Matthai Philipose
Intel Research Seattle
1100 NE 45th Street
6th Floor
Seattle, WA 98105
matthai.philipose@intel.com

Abstract

The use of large quantities of common sense has long been thought to be critical to the automated understanding of the world. To this end, various groups have collected repositories of common sense in machine-readable form. However, efforts to apply these large bodies of knowledge to enable correspondingly large-scale sensor-based understanding of the world have been few. Challenges have included semantic gaps between facts in the repositories and phenomena detected by sensors, fragility of reasoning in the face of noise, incompleteness of repositories, and slowness of reasoning with these large repositories. We show how to address these problems with a combination of novel sensors, probabilistic representation, web-scale information retrieval and approximate reasoning. In particular, we show how to use the 50,000-fact hand-entered Open-Mind Indoor Common Sense database to interpret sensor traces of day-to-day activities with 88% accuracy (which is easy) and 32/53% precision/recall (which is not).

Introduction

A system that can track the state of the world at multiple levels as humans go about their day-to-day activities is of interest both for conceptual and practical reasons. Conceptually, the ability to recognize and reason about what activities a person is doing, what the resulting physical state of the world is, what the likely emotional state of the actors is, and so on, is at the heart of computational models of human intelligence. From a pragmatic viewpoint, a whole variety of tasks such as caregiving, security monitoring, training and directing, which are currently considered expensive "high-touch" jobs that depend solely on humans become amenable to automated support if the computer can reason about the world. Researchers have recognized a variety of factors preventing this level of reasoning in machines, especially the need for very large quantities of common sense (McCarthy 1996; Minsky 2000), for noise-resistant representations and reasoning, and for very large quantities of labeled data connecting sensor signals to symbols. In this paper, we show how to leverage recent advances in very large scale common sense representation, information retrieval, statistical inference and sensing to build an automated system that can track the day-to-day world at multiple levels.

The most ambitious sensor-based day-to-day state estimation systems to date have been human activity recognition systems. Using sensors such as accelerometers and coarse audio (Bao & Intille 2004; Lester *et al.* 2005), these systems have been able to do an excellent job of recognizing simple activities such as walking, running and climbing stairs. On more complex activities, *e.g.,* various cooking, cleaning and personal grooming tasks, traditional general-purpose systems (typically based on vision (Moore, Essa, & Hayes 1999; Duong *et al.* 2005), potentially in concert with the above sensors) have had more limited success for two particular reasons. First, it has proved extremely difficult to detect salient high-level features (such as objects in use) robustly under day-to-day conditions. Second, acquiring models for the activities has proved difficult because of the need to acquire very large quantities of labeled data under a variety of circumstances.

A promising development in feature detection is the emergence of *dense sensors*, based on Radio Frequency Identification (RFID) and other wireless technology that can detect robustly the use of even small objects like toothbrushes. Using simple object-use-based models of activities, systems based on these sensors (Philipose *et al.* 2004; Tapia, Intille, & Larson 2004) have been able to detect a large variety of high-level activities robustly with high accuracy. The use of simple object-use models allows an interesting solution to the problem of acquiring models for activities (Perkowitz *et al.* 2004; Wyatt, Philipose, & Choudhury 2005). Since these models essentially capture the correlation between activity names (*e.g.*, "make coffee") and object names (*e.g.*, "mug", "spoon"), and the mapping represents common sense (*i.e.*, most people use the most of the same objects), it is possible to mine them using term-correlation on a large generic corpus such as the web. The weak classifiers so obtained can serve as priors for unsupervised learning on *unlabeled* data that can improve the model automatically.

It is intriguing to adapt the idea of using dense sensors to facilitate common sense activity recognition to the problem of recognizing generic day-to-day state using very large scale common sense. The adaptation is challenging for a va-

Copyright © 2006, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

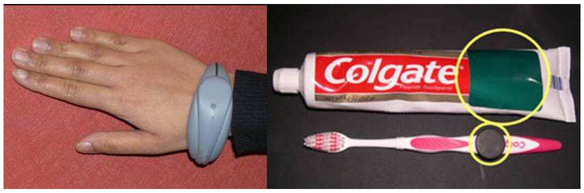

Figure 1: The iBracelet (left) and RFID tags (right)

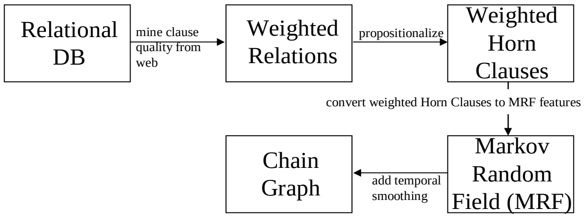

Figure 2: The SRCS data conversion architecture

riety of reasons. First, it is unclear how to represent models: common sense knowledge relating aspects of the world state tends to be expressed as declarative relations; the HMMs used for representing activities seem inadequate to the task. Second, it is unclear that term occurrence statistics are a practical means of acquiring arbitrary common sense information from the web. Third, given that we expect both the number of state variables of interest and the relationships between them to be very large, and given that we want to track state over time, it is unclear how to perform learning and inference efficiently.

In this paper, we present the architecture and implementation of a system called SRCS (for State Recognition using Common Sense) that solves the above problems. SRCS represents information as chain graphs (Buntine 1995), a factored probabilistic graphical representation that allows both directed and undirected dependences. It combines human-entered declarative relational databases of commonsense (Singh *et al.* 2002b; Lenat & Guha 1990) with web-wide information retrieval techniques based on lightweight syntactic analysis (Brill *et al.* 2001; Etzioni *et al.* 2004) to produce chain graphs representing the databases. It exploits the structure of the graph to introduce simple but highly effective techniques for performing inference fast on the extremely large graphs generated. We show that when reasoning about 75 minutes of day-to-day activity data from two subjects at a 2.5-second granularity, SRCS is able to track various aspects of the world with accuracy, precision and recall of 88%, 32% and 53%, far above that of baseline schemes.

To our knowledge, SRCS is the first system to show how large common sense databases can be used to interpret sensor data collected about the broad state of the world.

Sensors

Figure 1 shows the iBracelet (Fishkin, Philipose, & Rea 2005), the sensor we use to detect object use during an activity. The iBracelet works as follows. RFID tags are attached to objects (*e.g.*, the toothbrush and tube of toothpaste in Figure 1) whose use is to be detected. These tags are small, 40-cent battery-free stickers with an embedded processor and antenna. The bracelet issues queries for tags at 1Hz or faster. When queried by a reader up to 30cm away, each tag responds (using energy scavenged from the reader signal) with a unique identifier; the identifier can be matched in a separate database to determine the kind of object. The bracelet either stores timestamped tag IDs onboard or transmits read tags wirelessly to an ambient base station, and lasts 12 to 150 hours between charges. We assume that if a bracelet detects a tagged object, the object is in use, *i.e.*, hand proximity to objects implies object use. This definition leads to noise in object-use data, since objects not in use may be near the hand, and some objects may be grasped too far from a tag. Because this noise is low, we assume below that the bracelet yields a stream of the names of objects used in the current task. In its current avatar, the iBracelet makes it feasible to track the use of thousands of object in a household.

Data Acquisition and Representation

Figure 2 shows how SRCS acquires and transforms the representation of commonsense facts that represent its world view. SRCS operates by collecting sensory input from a user and employing statistical inference methods to reason about various predefined facts about the state of the world (*e.g.*, “Is the light on?”, “Is the user in the kitchen?”, “Is the user hungry?”). The model translating between observations and abstract state is acquired from existing hand-created commonsense databases, weighted by quality using automated web-based information retrieval techniques, translated into logical form, converted into a non-temporal probabilistic graphical model (PGM) to enable consistent reasoning under uncertainty, and finally converted into a temporal PGM well suited for efficient inference over time. Below, we describe each of these components in detail.

Common Sense Databases

We obtain the basic facts that we reason about, and the relationship between them from the Open Mind Indoor Common Sense (OMICS) (Gupta & Kochenderfer 2004) database. Similar to CyC (Lenat & Guha 1990), OMICS is a user-contributed database, based on the interface described by (Singh *et al.* 2002a), except that unlike CyC, which has a small dedicated team of humans adding facts, OMICS allows users from all over the Internet to add facts. Users are presented with fill-in-the-blank questions such as “You *blank* when you are *blank*”, with the expectation that users will fill in suitable words/phrases (*e.g.*, “eat” and “hungry”) in the two blanks. The sentence templates map into relations, *e.g.*, the `people(Action,Context)` relation, which may contain the instance `people(eat,hungry)`. Figure 3 shows some other instances in the database. SRCS uses roughly 50,000 such instances spanning 15 relations.

```
actiongeneralization('investigate cause
of','alarm','smoke alarm')
actiongeneralization('wipe off','floorcover','carpet')
actiongeneralization('clean','floorcover','carpet')

                              ...
contextactions('full garbage bag','put the garbage
in','trash')
contextactions('making toasted bread','slice','bread')


                              ...
people('eat','are hungry')
people('drink water','are thirsty')
```

Figure 3: Sample facts from OMICS. The predicate "`actiongeneralization`" represents generalized versions of objects used in actions. "`contextaction`" associates a context with specific actions. "`people`" associates states of a person with actions performed by that person.

Gauging the Quality of Facts

Given that the data in the OMICS database is contributed by (non-dedicated) humans, it contains a number of nominal "facts" that do not represent common sense. The database as it stands provides no information on the degree to which individual relations should be trusted. We use the lightweight web scale syntactic analysis techniques from the KnowItAll (Etzioni *et al.* 2004) system to estimate the degree to which each relation should be trusted.

KnowItAll is an information retrieval system designed to extract and evaluate widely known facts (and therefore also common sense) from the web. At the heart of KnowItAll is a template-based system that works as follows. To evaluate instances of a particular relationship, say `people(Action,Context)`, KnowItAll uses a small number of examples of the relation to induce a number of text templates that exemplify the relation. For instance, it may induce the two templates `"[action] when * [state]"` and `"[state] * [action]"` because, for instance phrases such as `"eat when hungry"` appear on the web. Using normalized counts of incidences of these patterns in the web, KnowItAll is able to produce a measure of how reliable a particular proposition may be.

A limitation of the base KnowItAll system is that it does not directly handle assessment for predicates with more than three arguments. Thus, the four-place `stateChange(Action,Object,State,State)` cannot be processed directly. In this case, we use a set of three-place templates, `"[action] * [state1] [object]"`, `"[state2] because * [action]"` and `"[state2] due to [ing_form(action)]"` to assess it. Each template corresponds to a Boolean feature: if the number of hit counts obtained by instantiating it is greater than $c = 5$, the feature is true, else false. The features are combined to give the final score for the whole template. Although this technique does not always work well, it works well for identifying very reliable propositions.

The end result of this pass is a database with relation instances $r_1, \ldots, r_n$ with a weight w_i associated with r_i.

Converting from Relational to Propositional Form

We convert the relational entries in the common sense databases into propositional form such that the atoms of the propositional form correspond to observables and propositions we wish to reason about.

SRCS's propositions are Horn clauses of the form $\mathrm{p}_1 \wedge \ldots \wedge \mathrm{p}_N \Rightarrow \mathrm{p}_{N+1}$, where the $\mathrm{p}_1, \ldots, \mathrm{p}_N$ are either *constants* or *atoms* and p_{N+1} is an atom. Constants are of five types: `object`, `action`, `location`, `context`, and `state`. Types of atoms include (all told, SRCS uses 8 atoms):

- `useInferred(O)` — object `O`'s use has been observed, or indirectly inferred. This is a key atom, since it binds to sensor observations and grounds our inference.
- `stateOf(O,S)` — object `O` is currently in state `S`.
- `locationInferred(L)` — the current location of the user is `L`.
- `personIn(S)` — the user is in state `S` (*e.g.*, "sleeping", or "happy").
- `actionObserved(A)` — the action `A` is observed in the user's world..

We convert from individual relational entries to corresponding Horn clauses using a small fixed set of rewrite rules (approximately 20). For example, one such rule is: `people(S,A)` $\rightsquigarrow$ (`actionObserved(A)` $\Rightarrow$ `personIn(S)`). Thus if OMICS contains the fact `people(angry,yell)`, then we define the atoms `actionObserved(yell)` and `personIn(angry)`, and add them to our set of atoms. Weights from the incoming relations are preserved during rewrites so that the Horn clauses are weighted.

The rewrite rules encode many assumptions about what the relations in the database mean. For instance, we provide a purely propositional view of the world: we cannot quantify over multiple instance. Further, each proposition is assumed to refer to the state of the world in a single time slice. Although these assumptions may not always be true, we believe that the end result is still of value.

From Weighted Clauses to Markov Random Fields

Say we wish to track information about the state of the environment over a series of time intervals $1, 2, ...T$. We are given a set of objects $O_1, ...O_n$ whose use may be tracked over this time, and a set of atoms $f_1, ...f_m$ which we wish to track over this time. Let $o_{i,t}$ be the random variable representing the use of object O_i at time slice t. Let $f_{i,t}$ be the random variable representing the truth value of atom f_i at time slice t. We will model the probability $p(f_{1,t}, ...f_{m,t}, o_{1,t}, ...o_{n,t})$. of the world for a given time slice t using a *Markov Random Field (MRF)*. A MRF consists of a graph whose set of vertices V — in this case all $f_{i,t}$ and $o_{i,t}$ for a given t — are connected by a set of cliques $c_i \subset V$. Each c_i has a *potential functions* ϕ_i mapping assignments of c_i to nonnegative reals. For an assignment

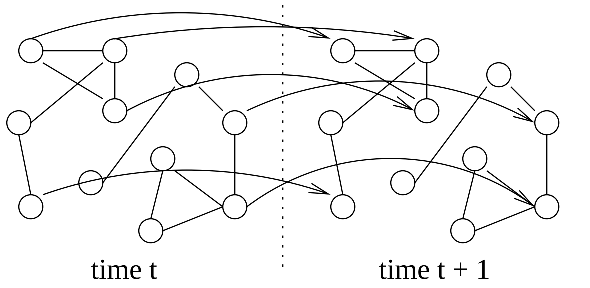

Figure 4: The chain graph representing our model over time

$\mathbf{f}_t, \mathbf{o}_t$ to $f_{1,t}, ...f_{m,t}, o_{1,t}, ...o_{n,t}$, we have

$$p(\mathbf{f}_t, \mathbf{o}_t) = \frac{1}{Z} \exp(\sum_i \lambda_i \phi_i(c_i(\mathbf{f}_t, \mathbf{o}_t)))$$

where $c_i(\mathbf{f}_t, \mathbf{o}_t)$ is the assignment imposed on c_i by $\mathbf{f}_t, \mathbf{o}_t$ and Z is a normalization constant. λ_i is a weight placed on the clique c_i and represents a tunable parameter. For our current purposes, all λ_i are set to 1.

To represent the state of world $\mathbf{f}_t$ given the observations $\mathbf{o}_t$, we convert the weighted Horn clauses to an MRF as follows. For each atom and constant in the clauses, we have a node in the MRF. For each weighted clause $\mathrm{p}_1 \wedge \ldots \wedge \mathrm{p}_N \Rightarrow \mathrm{p}_{N+1}$ with score w, create a clique c of the nodes n_i corresponding to the p_i, and associate with it the potential $\phi(\mathrm{p}_1 \ldots \mathrm{p}_{N+1}) = w$ if $\mathrm{p}_1 \wedge \ldots \wedge \mathrm{p}_N \Rightarrow \mathrm{p}_{N+1}$, and $1 - w$ otherwise. In other words, we favor joint assignments that satisfy as many clauses as possible. The actual potential we use is slightly different so as to discourage assignments that set the atoms on the left of the clauses being set to false (since this will make clauses trivially true). This technique of converting weighted logical formulas to MRFs is similar to that used, for instance, in Richardson & Domingos(2006).

Temporal Dependences via Chain Graphs

The MRF is an effective representation for the relationships between observations and propositions about the world for a moment in time. However, we also wish to incorporate temporal relationships and infer over periods of time. One way to do this is to employ a *dynamic MRF*, in which we create an MRF for each time slice, then connect them into a larger MRF with potential functions between time slices. For our purposes, however, a dynamic MRF requires an immense amount of computation and is too inefficient; to infer over one slice of time t, we must infer over all time slices at once. A better model would permit us to calculate probabilities at time t conditioned only upon those probabilities at time $t-1$ or $t+1$, not unlike the technique of rollup filtering in dynamic Bayesian networks. We thus employ a different technique for incorporating temporal relationships into our model.

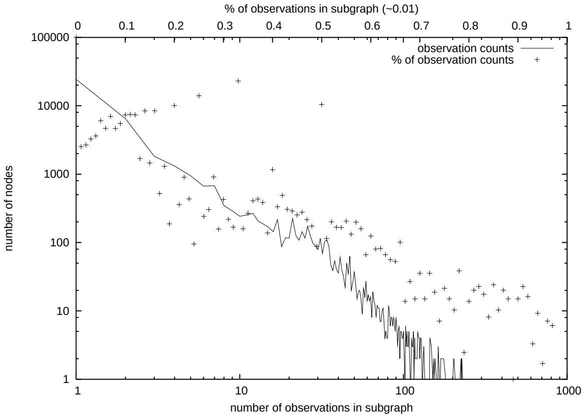

Figure 5: Number of observations, and percentage of total observations, in selected subgraphs of depth $d = 2$.

SRCS makes use of a *chain graph* (CG), described in detail in (Buntine 1995). A chain graph is a hierarchical combination of directed and undirected graphical models. To produce a temporal model, we produce a chain graph in which a series of MRFs, each representing a time slice t, are linked by directed edges, representing conditional probabilities between nodes in different slices. If f_t and f_{t+1} are nodes in time slices t and $t+1$, respectively, a directed edge (f_t, f_{t+1}) indicates that random variable f_{t+1} is dependent upon f_t.

A model of the resulting CG is depicted in Figure 4. As described in (Buntine 1995), there is a natural decomposition of this graph into component Random Fields and Bayesian networks. The form of our graph allows for a natural expression of all dependencies present in a single time slice, including those from an adjacent time slice, as a MRF. We will describe filtering (*i.e.*, forward inference) here. We may calculate the distribution of a node f_t in the CG as $P(f_t) = \sum_{v \in DPa(f_t)} P(f_t|v)P(v)$, where $DPa(f_t)$ is the set of *directed parents* of f_t — *i.e.*, all nodes v_{f-1} such that v_{f-1} has a directed edge to f_t; in our current model, $DPa(f_t)$ is either empty or $\{f_{t-1}\}$.

We assume that an atom's truth value degrades at a constant rate, absent other observations, *i.e.*, if f_t is true at time t, then $P(f_{t+k}) = p_T^k$ for some p_T, and if f_t is false at t, then $P(f_{t+k}) = p_F^k$ for some p_F. We fix p_T to .95 and p_F to .095, since we consider propositions more likely to be false than true. This is our model for defining $p(f_{t+1}|f_t)$ for each proposition f_t.

Inference

Say that we wish to track the truth values of propositions over a period of time $1...t$. To infer state at a time i, we will fix those nodes whose truth value is known for a time slice i — in this case, all propositions of the form `useinferred(O)` will be set to true or false depending on whether use of object O was detected in time slice i. Using the marginals for propositions at time slice $i - 1$, we may then perform inference on this graph to calculate the probabilities of unknown variables at time i. We use loopy belief propagation (BP), as described in (Pearl 1988), for inference.

Since the OMICS data is provided by untutored users in natural language form, the propositions and observations produced in its processing sometimes be difficult to resolve with observed data. We use a basic synonym-based matching scheme (*e.g.*, matching "bucket" with "pail"), as well as other minor ad hoc techniques, to match the observed uses of objects with the appropriate propositions.

Query-Directed Pruning

This model still poses a problem: representing all the known commonsense propositions about the world at a particular time t can require a huge graph. After processing the OMICS database, our graph contains 55000+ nodes in a single time slice for representing both observations and hidden variables; inference on each time slice (where each slice represents 2.5 seconds of data) using loopy BP takes approximately 30 minutes. We thus employ techniques to make inference over this graph more tractable.

It is unnecessary to performing inference over the entire graph of variables for two reasons. First, many of the variables may be irrelevant to the current context in which the system is being used; the state of the bathroom sink, for instance, is unlikely to be of interest when observing activities in the garage. Second, we typically use a *query set* of variables whose truth values we are interested in. Instead of performing inference over the entire graph, we perform inference over a smaller subgraph which includes all possible observations and the query variables we wish to track, in addition to many others.

In practice, we do this by selecting the k variables we wish to track and selecting every node of distance d from each proposition in the time slice MRF (we used $d = 2$). To insure all observations were included in the graph, we then find the shortest path from each proposition to the node `useinferred(O)` for each object O being tracked by our system, if such a path exists, and select all nodes along these paths as well. The selected nodes, and the potentials between them, comprise the subgraph which we perform inference over. This is the method we will use in our experiments to track the state of a subset of variables over time.

Figure 5 gives some insight into why the query-based pruning technique with $d = 2$ may work for our graph. Many nodes have a grounded observation node within their 2-neighborhood, and quite a few nodes have two or more such nodes. Note that if variables such as locations and primitive actions (such as limb movement) were directly observable and ground, the density would increase, further favoring pruned inference.

Output Thresholding

The probabilities that SRCS outputs, in its current form, do not have much use as an *absolute* measure of the probability of an action. However, we have found they are useful as a *relative* measure, as they tend to increase as the likelihood of an activity increases. We thus use simple machine learning techniques to identify "threshold" probabilities beyond which we label a proposition true or false. To label traces, we feed SRCS object traces labeled with ground truth for the variables being tracked, and train decision stumps on each proposition to recognize the optimal threshold value for labeling variables. We then perform inference over object traces via the technique described, with observations of object use fixed to true or false, and label according to whether the probabilities output fall above or below the learned thresholds.

brush teeth	take medication	water plants
shave your face	take a shower	watch television
dust shelves	eat cereal	groom hair
write a letter	make cereal	wash windows
	make tea	

Figure 6: The set of activities for which experimental data was collected.

a	`actioninferred(brush teeth with)`	**b**	`contextinferred(brush teeth)`
c	`stateof(toothbrush,wet)`	**d**	`stateof(teeth,clean)`
e	`stateof(duster,dirty)`	**f**	`locationinferred(shower)`
g	`stateof(cereal,prepared)`	**h**	`actioninferred(eat)`
i	`locationinferred(kitchen)`	**j**	`likelyaction(shower with)`
k	`locationinferred(bathroom)`	**l**	`actioninferred(write)`
m	`personin(want to be entertained)`	**n**	`actioninferred(add milk to)`
o	`stateof(cereal,eaten)`	**p**	`likelyaction(swallow)`
q	`actioninferred(prepare tea in)`	**r**	`stateof(tea kettle,hot)`
s	`actioninferred(shave using)`	**t**	`stateof(window,dirty)`
u	`actioninferred(sit)`	**v**	`actioninferred(write a letter)`
w	`locationinferred(pantry)`	**x**	`locationinferred(greenhouse)`

Figure 7: Variables tracked during inference.

Evaluation Methodology and Results

For our experimental evaluation, we collected traces of the iBracelet's output in an experimental setting as worn by three users while performing various daily activities in a simulated home environment. The list of activities performed can be seen in Figure 6. A total of 5-7 minutes worth of performance of each activity was collected, for a total of approximately 70-75 minutes of data. These traces were divided into time slices of 2.5 seconds; reasoning was to be performed over each of these time slices.

For these activities, we considered a variety of variables about the state of the world which could be relevant to these activities. We then selected a set of 24 Boolean variables in the collected SRCS database which represented these variables, or were semantically very close to them; these variables are listed in Figure 7. We then recorded their "truth"

Model	Accuracy	Precision	Recall
Random	50.00%	8.04%	50.00%
All labeled false	92.55%	-	0.00%
SRCS/uniform prob.	80.26%	18.07%	46.44%
SRCS/KnowItAll prob.	88.42%	31.73%	53.07%

Figure 8: Per proposition and mean accuracy, precision, and recall.

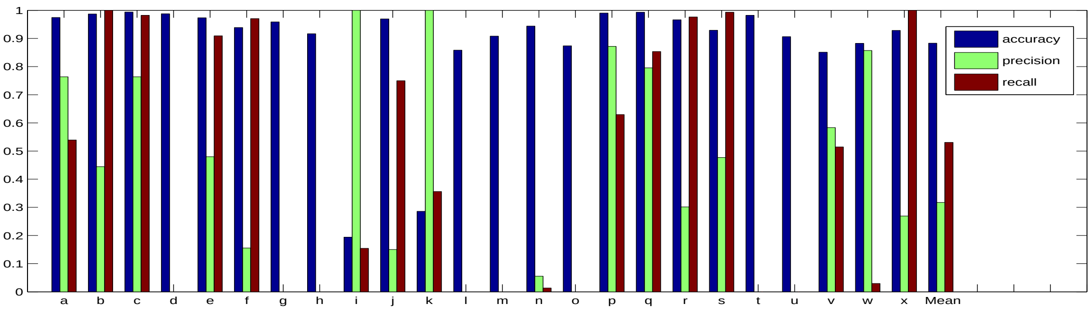

Figure 9: Per proposition and mean accuracy, precision, and recall measures as labeled by SRCS.

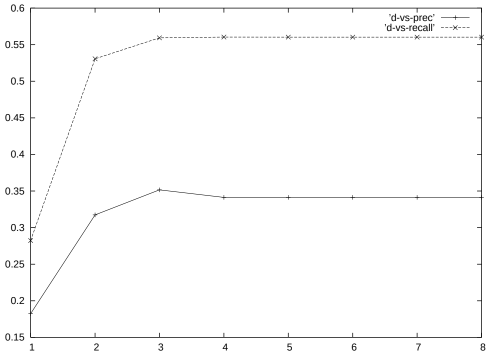

Figure 10: Plot of precision and recall versus increased pruning depth (d).

value as being true or false for each interval of time in the trace. In some instances, the labeling of the variables involved were somewhat subjective in nature; this is a natural consequence of the OMICS database being collaborative, unsupervised, and represented by natural language. However, we are interested in observing how closely the model may track the human interpretation of these variables, so we consider a human labeling of these traces to be an appropriate source of ground truth.

We ran SRCS, and trained its stumps on a sampling of data for each activity ($\sim$20 min total). We then found the labeled truth values it provided as per the method previously described, and compared them to the given "ground truth" values provided by human labeling. This was done in two runs: one with the potentials imposed by the OMICS data all set to a uniform strength, and one with the strengths defined by the mining by KnowItAll. Note that most of the variables are false most of the time ($\sim$94% of labels are false), and finding true variables is of greater interest; we thus consider the standard IR measures of precision and recall with respect to discovery of true variables in addition to labeling accuracy.

We are also interested in observing the effect of different choice of query-directed pruning depth d on the resulting output. We thus ran the experiments for values of d between 1 and 8 (the total diameter of the graph produced), to observe the difference.

Total mean accuracy, precision, and recall measures for all variables combined are compared in Figure 8. Results for labeling of each proposition upon traces when KnowItAll mined potentials are used are given in Figure 9, and compared to random labeling and labeling all variables as false. While the precision and recall figures may seem low, it is worth noting that they perform far better than the baseline strategies, and have been achieved with the use of mined and preexisting data and little developer effort in developing the actual model.

We see that use of the mined quality scores considerably improves the accuracy and precision. Much of the difference in results is caused by a decrease in positive labels (*i.e.*, variables labeled "true"); while this lowers the recall slightly, the precision is considerably improved by the decrease in false positives. The mined scores appear to be effective in weeding out correlations of lower quality in the OMICS data.

Precision and recall are quite high on certain variables, but performance on these measures is quite inconsistent, being very high on certain variables and low on others. Part of this is due simply to the inadequacies of the OMICS database; because OMICS is represented in natural language and built in an informal fashion, there may be many "holes" in its representation of the world. These deficiencies in the model could be resolved through human entry of correlations between variables or, preferably, automatic discovery of such variables via mining, machine learning, or other means.

Finally, a graph comparing the precision and recall with increasing values of the parameter d can be seen in Figure 10. We see that improvement in precision and recall levels off for a remarkably small value of the pruning depth: above $d = 3$, recall is nearly flat, and precision even drops slightly. We chose $d = 2$ for our experiments because the precision and recall are nearly as good, but inference time is significantly less, because the number of nodes in the pruned graph grows exponentially with the pruning depth.

Future Work

While this work provides an interesting architecture for inference about the state of the world with respect to everyday activity, there remains considerable room for improvement. For instance, our initial model is built in a relatively simple manner; the use of learning weights on the graph potentials with labeled object traces is likely to considerably improve inference accuracy. While labeling traces in this manner could represent considerable cost in terms of human effort, there may be much promise in semi-supervised learning methods like those described in (Zhu 2005), in which a small amount of labeled traces may be used with a large set of unlabeled traces, which can be collected with minimal effort by simply letting the system passively record activities for some period of time. We are currently exploring effective learning of the weights and potential functions for the graphical model described, as well as making effective use of mined information like that provided by systems like KnowItAll to minimize the need for labeled data to learn from.

We are also currently exploring the integration of other sources of sensory input into our system; while a trace of object use can be useful on its own in recognition and analysis of activity, other input, such as tracking of movement, can be helpful as well. To this end, we hope to incorporate input from the multi-sensor board described in (Lester *et al.* 2005), which measures a user's acceleration in each direction and ambient environmental information, into the input of our system. This input could be use to recognize instances of different actions occurring with objects, *e.g.*, the use of "chopping" with a knife.

Another promising subject of study is the problem of selecting the subset of variables about the world that are relevant to the user's context. In these experiments, we have only considered a fixed, predefined subset of variables as a means of selecting the variables to use in inference. A more sophisticated system could attempt to determine the current context of the user (*e.g.*, in the bathroom, making dinner, fixing the car, etc.), select the subset of variables that were relevant to this context, and perform inference over them. We wish to explore solutions to this problem in future work as well.

Conclusions

Densely deployable wireless sensors developed in recent years have made it possible to detect objects used in daily activities in great detail. We show in this paper that when coupled with recent advances in collaborative common sense databases, web-scale information retrieval and large-scale statistical inference, these sensors can yield a system capable of tracking the state of the world at various levels of details with relatively little human effort.

This work suggests many future directions, including the use of other handmade databases, the use of information retrieval techniques to supplement these databases, the use of other dense sensors such as accelerometers, audio and GPS to ground even more commonsense nodes, an exploration of scaling inference in sound ways, learning parameters and structure of the huge network, the use of richer models such as first-order models, and testing the resulting systems on much larger amounts of data.

References

Bao, L., and Intille, S. 2004. Activity recognition from user-annotated acceleration data. In *Proc. of PERVASIVE 2004, LNCS 3001*, 1–17.

Brill, E.; Lin, J. J.; Banko, M.; Dumais, S. T.; and Ng, A. Y. 2001. Data-intensive question answering. In *TREC*.

Buntine, W. 1995. Chain graphs for learning. In *UAI 1995*.

Duong, T. V.; Bui, H. H.; Phung, D. Q.; and Venkatesh, S. 2005. Activity recognition and abnormality detection with the switching hidden semi-markov model. In *CVPR*, 838–845.

Etzioni, O.; Cafarella, M.; Downey, D.; Popescu, A.; Shaked, T.; Soderland, S.; Weld, D.; and Yates, A. 2004. Methods for domain-independent information extraction from the web: An experimental comparison. In *Proc. of 19th Annual AAAI Conference*.

Fishkin, K. P.; Philipose, M.; and Rea, A. 2005. Hands-on RFID: Wireless wearables for detecting use of objects. In *ISWC 2005*, 38–43.

Gupta, R., and Kochenderfer, M. J. 2004. Common sense data acquisition for indoor mobile robots. In *AAAI*, 605–610.

Lenat, D., and Guha, R. V. 1990. *Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project*. Addison-Wesley.

Lester, J.; Choudhury, T.; Kern, N.; Borriello, G.; and Hannaford, B. 2005. A hybrid discriminative/generative approach for modeling human activities. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*.

McCarthy, J. 1996. From here to human-level AI. In *Proc. of Principles of Knowledge Representation and Reasoning (KR)*.

Minsky, M. 2000. Commonsense-based interfaces. *Communications of the ACM* 43 No. 8:67–73.

Moore, D. J.; Essa, I. A.; and Hayes, M. H. 1999. Exploiting human actions and object context for recognition tasks. In *ICCV*, 80–86.

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kauffman.

Perkowitz, M.; Philipose, M.; Patterson, D.; and Fishkin, K. 2004. Mining Models of Human Activities from the Web. In *WWW 2004*.

Philipose, M.; Fishkin, K.; Perkowitz, M.; Patterson, D.; Kautz, H.; and Hahnel, D. 2004. Inferring activities from interactions with objects. *IEEE Pervasive Computing Magazine* 3(4):50–57.

Richardson, M., and Domingos, P. 2006. Markov logic networks. *Machine Learning* 62:107–136.

Singh, P.; Lin, T.; Mueller, E. T.; Lim, G.; Perkins, T.; and Zhu, W. L. 2002a. Open mind common sense: Knowledge acquisition from the general public. In *CoopIS/DOA/ODBASE*, 1223–1237.

Singh, P.; Lin, T.; Mueller, E. T.; Lim, G.; Perkins, T.; and Zhu, W. L. 2002b. OpenMind: knowledge acquisition from the general public. In *Proceedings of the First International Conference on Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems*.

Tapia, E. M.; Intille, S. S.; and Larson, K. 2004. Activity recognition in the home using simple and ubiquitous sensors. In *Pervasive*, 158–175.

Wyatt, D.; Philipose, M.; and Choudhury, T. 2005. Unsupervised activity recognition using automatically mined common sense. In *AAAI*, 21–27.

Zhu, X. 2005. Semi-supervised learning literature survey. *Computer Sciences TR 1530, University of Wisconsin, Madison*.