Improving Predictions using Aggregate Information

Amit Dhurandhar
adhuran@us.ibm.com
Mathematical Sciences Dept.
IBM T.J. Watson
1101 Kitchawan Road
Yorktown Heights, USA

ABSTRACT

In domains such as consumer products or manufacturing amongst others, we have problems that warrant the prediction of a continuous target. Besides the usual set of explanatory attributes we may also have exact (or approximate) estimates of *aggregated targets*, which are the sums of disjoint sets of individual targets that we are trying to predict. Hence, the question now becomes can we use these aggregated targets, which are a coarser piece of information, to improve the quality of predictions of the individual targets? In this paper, we provide a simple yet provable way of accomplishing this. In particular, given predictions from any regression model of the target on the test data, we elucidate a provable method for improving these predictions in terms of mean squared error, given exact (or accurate enough) information of the aggregated targets. These estimates of the aggregated targets may be readily available or obtained – through multilevel regression – at different levels of granularity. Based on the proof of our method we suggest a criterion for choosing the appropriate level. Moreover, in addition to estimates of the aggregated targets, if we have exact (or approximate) estimates of the mean and variance of the target distribution, then based on our general strategy we provide an optimal way of incorporating this information so as to further improve the quality of predictions of the individual targets. We then validate the results and our claims by conducting experiments on synthetic and real industrial data obtained from diverse domains.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Data Mining

General Terms

Algorithms

Keywords

Regression,Hierarchical,Coarse to fine

KDD'11, August 21–24, 2011, San Diego, California, USA.
Copyright 2011 ACM 978-1-4503-0813-7/11/08 ...$10.00.

1. INTRODUCTION

In many industries such as consumer products, manufacturing where we have a supply chain consisting of a manufacturer who produces and sends goods to various distribution centers (DC) who further redistribute the goods amongst the stores, we observe a certain delay from the time that the goods are produced to the stores finally receiving them. As time goes by, finer and finer pieces of information become available. From a strategic point of view, however, the manufacturer may want to know initially how his goods are going to be distributed among the various DCs and stores with as much accuracy as possible. Based on his past experience, he may be able to come up with predictions of how much each store or DC might order. However, the question we ask is the following: is it at all possible to improve the quality of these predictions knowing the total amount of goods that are going to be distributed for the current time period? In this paper, we answer this question affirmatively, that is, we provide a simple method to provably improve predictions obtained based on past observations, by using estimates or actual values of the target at a coarser level of granularity for the current time period.

It is easy to see that if we have the true values (or accurate estimates) at a particular level of granularity, we can sum them up to get estimates at a coarser level of granularity. For example, if we have predictions for a coarser level of granularity and true values at a finer level, we can improve the predictive accuracy at the coarser level by aggregating the finer estimates and using them as predictions. This is seen in figure 1a. However, if we have the converse problem then the solution is not obvious. What we mean by this is that, if we knew coarser values and were trying to improve the quality of our predictions at a finer level, it's not clear if there is in fact a provable way of improving the accuracy. In other words, given predictions of the target for the current time period based on past experience – this could be the output of a regression model or something else – we improve (more precisely never worsen) the quality of these predictions using *aggregated target* information, i.e. using information about the sums of different sets of targets we are trying to predict. This will become clearer if we consider figure 1b, where we have predictions for the three datapoints (denoted by circles). The sum of the true targets is 9 and the method we suggest in this paper uses this value 9 to improve the accuracy of the predictions. In fact, as you will see, even if the value 9 is not the exact sum of the targets but an "accurate enough" estimate, our method still guar-

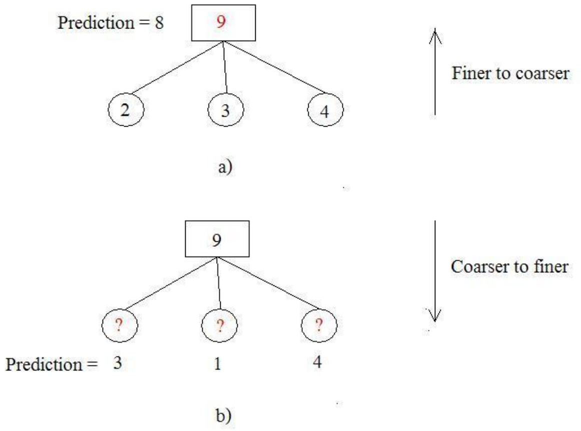

Figure 1: a) Using finer estimates to improve coarser predictions. b) Using coarser estimates to improve finer predictions.

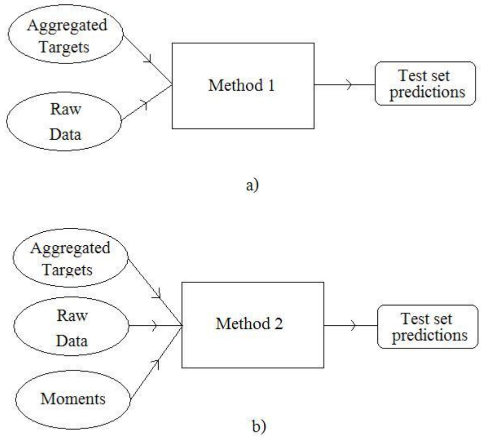

Figure 2: The basic method in this paper represented in a) takes as input the raw data and the aggregated targets (exact or estimates) at a single or multiple levels and outputs the test predictions. The enhanced method represented in b) takes in addition the moment information of the target distrbution to give further improved predictions.

antees that the new predictions obtained by its application will be no less accurate than the old predictions.

Such a method can be used not only for supply chain kind of problems but in any problem where good quality aggregate information is available and we want to predict at a finer level of granularity. A good example of this may be census data where information at a national or state level may be more reliable than data at a city or county level where there might be missing data since some people may not turn in the survey. Predicting the missing values can be done more effectively knowing the aggregate information. If aggregate information is available at multiple levels of granularity with varying accuracy, choosing the right level of granularity so as to maximize the improvement in predictions is not obvious. In this paper, we provide a criterion for choosing this optimal level of granularity.

Information at a coarser level of granularity may not always be available as is the case in standard machine learning settings. In this case, we could build regression models on the historical data by aggregating it at various levels of granularity and use the ”best” model to give us estimates of the aggregated targets at that level of granularity. These estimates can then be used in conjunction with our method to improve the predictions at the finest level of granularity which we care about. The ”best” model is not necessarily the most accurate model amongst those built at the various coarser levels of granularity since, as we will see later, the amount of improvement in predictive accuracy at the finest level by using our method is a function of both the accuracy of the models and the level of granularity they are built at. Consequently, we provide an algorithm for choosing the model that is most likely to elevate the accuracy of the predictions. In the trivial case, the best model might be the model at the finest level, which would suggest that aggregating the data isn't too helpful. An algorithm of this nature however, can be used for a wide variety of machine learning tasks such as predicting time series data where the aggregate models would predict the potential drift, if any, over time and this drift if accurately captured can assist in improving individual predictions [6, 1, 2]. Another example is microarray data which is sparse and hence aggregating it can help predictive accuracy [8].

In addition to this, we enhance our method of using aggregated targets to improve finer predictions to be able to use distribution information of the target if available. In particular, we find the optimal weighting based on the mean and variance of the distribution of the target that will maximize the impact on the quality of the predictions in expectation. If this information is not available one may estimate these moments from the data, if deemed appropriate. The input-output of the original method and the enhanced method are pictorially depicted in figure 2.

Using coarser information to improve predictions has been of some interest lately [9, 7, 10, 4, 11, 5]. In [9, 7, 4, 11] the authors employ this philosophy to improve performance of models in certain computer vision tasks (viz. pose estimation, face recognition, etc.). In [10] however, the idea is used to improve the performance of NLP models. The primary difference between this past literature and our work is that the results in this paper are for the regression setting while the past literature mainly considers the classification and the structured prediction setting. In the regression setting, there has been some work [5] related to using aggregate information, but this is with respect to a specific problem of identifying socio-economic factors that may lead to hospital admissions for heart and circulatory diseases. Our method in contrast is not necessarily restricted to any particular domain. In fact, as we will see in the experimental section, it is applicable in multiple diverse domains. Moreover, in this paper, we explicitly provide a provable method to improve predictions at a finer level of granularity using aggregate information.

The rest of the paper is organized as follows: In the next section, we first describe our method. We then formally state this through lemmas and theorems (proofs in appendix) that show that our method in fact works. We then show that the predictions can be further improved if the mean and variance of the target are accurately known. Based on the proofs of these previous results, in cases where we may have estimates at multiple (coarser) levels of granularity, we provide a criterion that chooses a level that using our method is most likely to maximize the improvement in the quality of predictions at the finer level of granularity. In traditional settings where these estimates may not be available apriori, we suggest an algorithm where we build regression models at multiple levels of granularity on the training set in order to obtain the corresponding estimates. We then using our criterion decide on the level and hence, the regression model to be used to improve the quality of predictions obtained from a regression model built at the (finer) level of granularity that we care about. In Section 3, we present results of experiments performed on synthetic and real datasets and empirically validate the efficacy of our method. In Section 4, we discuss further extensions and summarize the major findings in this paper.

2. TRICKLING DOWN AGGREGATES

In this section we first describe a simple method to trickle down aggregate information in order to improve finer predictions. We formally state the relevant results with proofs provided in the appendix. If estimates of the aggregated target are available at multiple (coarser) levels of granularity, we then based on our proofs for the previous results, suggest a criterion to choose the level that using our method is most likely to maximize the improvement in the quality of predictions at the (finest) level of granularity we want to predict. If these estimates are not available, we suggest an algorithm for obtaining them and using the criterion to decide the appropriate level.

Before we start formally describing our results we define a couple of terms.

Aggregation granularity: We define aggregation granularity k as the number of values at the finest level of granularity (i.e. at the level of the original dataset) summed together to form coarser estimates. For example, in figure 3a, the aggregation granularity is 2 since, if the circles represent datapoints at the finest level then the rectangles which denote coarser estimates are sums of pairs of these circles. Similarly, in figure 3b, the aggregation granularity is 3, since the rectangles are sums of triplets of the circles.

Aggregated targets: These are sums of the individual targets in each set, where the sets form a partitioning of the individual targets in the dataset. Note that the rectangles in figure 3 would denote aggregated targets if the circles denoted individual target values.

2.1 Method and Results

An informal description of our method where the aggregation granularity is k is as follows. We first sum up the various sets of k predictions corresponding to the aggregated targets which are already available or obtained by techniques described before. With this we have each of the aggregated targets associated with its own sum of k predictions. We now subtract each of these sums from the corresponding aggregated targets which gives us the corresponding differences. We then divide each of these differences by k and uniformly add them to the corresponding k predictions. Thus, in figure 1b, where 9 is the aggregated target with 3, 1 and 4 being the predictions, we would first add 3, 1 and 4 which gives us 8, then subtract 8 from 9 which gives us 1 and then finally add $\frac{1}{3}$ to the original predictions which would give us $\frac{10}{3}$, $\frac{4}{3}$ and $\frac{13}{3}$ as the new predictions. If additional information regarding the distribution (mean and variance) of the target is available, then rather than distributing the differences uniformly amongst the predictions an optimal convex weighting scheme is derived.

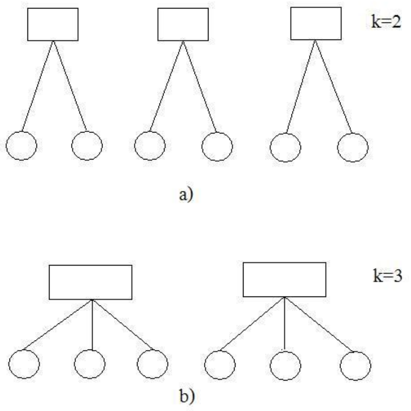

Figure 3: a) Aggregation granularity is 2. b) Aggregation granularity 3.

With this, we now present four results which includes a formal description of the method we described above.

- In Lemma 1 we show that knowing the *true or exact* values of the aggregated targets and predictions of the individual targets, our method can produce new modified predictions that are never worse in terms of mean squared error (MSE) than the original predictions.

- In Theorem 1 we show that knowing *approximate* values (within a certain error bound) of the aggregated targets and predictions of the individual targets, our method can produce new modified predictions that are never worse in terms of MSE than the original predictions.

- Lemma 2 shows that even if we know the exact values of the aggregated targets and have predictions of the individual targets, and if we alter our method slightly where we distribute the differences non-uniformly amongst the predictions, then the claim made in 1 no longer holds. In other words, the MSE of the new predictions might be greater than the old predictions if all the differences are not distributed uniformly.

- Lastly, in lemma 3 we show that in addition to knowing the aggregated targets and having predictions of the individual targets, if we also know the mean and variance of the target distribution then we can derive optimal weights for distributing the differences which may be non-uniform.

Lemma 1. *Consider two sets of N real numbers $X = \{x_1, x_2, ..., x_N\}$ and $\bar{X} = \{\bar{x_1}, \bar{x_2}, ..., \bar{x_N}\}$ (estimates). Let $A = \{a_1, ..., a_m\}$ and $\bar{A} = \{\bar{a_1}, ..., \bar{a_m}\}$ such that if, k is the aggregation granularity, $l_i = min(ik, N) - (i-1)k$, $m = \lceil \frac{N}{k} \rceil$, then $a_i = \sum_{j=(i-1)k+1}^{min(ik,N)} x_j$ and $\bar{a_i} = \sum_{j=(i-1)k+1}^{min(ik,N)} \bar{x_j}$. If $\epsilon_i = a_i - \bar{a_i}$ then,*

$$\sum_{j=1}^{N}(x_j - \bar{x_j})^2 \geq \sum_{j=1}^{N}(x_j - \hat{x_j})^2$$

where $\hat{x_j} = \bar{x_j} + \frac{\epsilon_{\lceil \frac{j}{k} \rceil}}{l_{\lceil \frac{j}{k} \rceil}}$

The result below shows that even if the values at the coarser level of granularity are not known exactly but with "some" error, they still can be used to enhance accuracy.

Theorem 1. *Consider two sets of N real numbers $X = \{x_1, x_2, ..., x_N\}$ and $\bar{X} = \{\bar{x_1}, \bar{x_2}, ..., \bar{x_N}\}$ (estimates). Let $A = \{a_1, ..., a_m\}$ and $\bar{A} = \{\bar{a_1}, ..., \bar{a_m}\}$ where if k is the aggregation granularity, then $l_i = min(ik, N) - (i-1)k$, $m = \lceil \frac{N}{k} \rceil$, $a_i = \sum_{j=(i-1)k+1}^{min(ik,N)} x_j$ and $\bar{a_i} = \sum_{j=(i-1)k+1}^{min(ik,N)} \bar{x_j}$. If $\epsilon_i = a_i - \bar{a_i}$ and $\delta_i \in [0, 2\epsilon_i]$ then,*

$$\sum_{j=1}^{N}(x_j - \bar{x_j})^2 \geq \sum_{j=1}^{N}(x_j - \hat{x_j})^2$$

where $\hat{x_j} = \bar{x_j} + \frac{\delta_{\lceil \frac{j}{k} \rceil}}{l_{\lceil \frac{j}{k} \rceil}}$

Lemma 2. *Consider two sets of N real numbers $X = \{x_1, x_2, ..., x_N\}$ and $\bar{X} = \{\bar{x_1}, \bar{x_2}, ..., \bar{x_N}\}$ (estimates). Let $A = \{a_1, ..., a_m\}$ and $\bar{A} = \{\bar{a_1}, ..., \bar{a_m}\}$ where if k is the aggregation granularity, $l_i = min(ik, N) - (i-1)k$, $m = \lceil \frac{N}{k} \rceil$, then $a_i = \sum_{j=(i-1)k+1}^{min(ik,N)} x_j$ and $\bar{a_i} = \sum_{j=(i-1)k+1}^{min(ik,N)} \bar{x_j}$. If $\epsilon_i = a_i - \bar{a_i}$ and $\forall i \sum_{j=(i-1)k+1}^{min(ik,N)} \alpha_j = 1$ where $\forall j \ \alpha_j \geq 0$ with all α_j (for any i) not being equal then there always exists a X and $\bar{X}$ such that,*

$$\sum_{j=1}^{N}(x_j - \bar{x_j})^2 \leq \sum_{j=1}^{N}(x_j - \hat{x_j})^2$$

where $\hat{x_j} = \bar{x_j} + \alpha_j \epsilon_{\lceil \frac{j}{k} \rceil}$

Lemma 3. *Consider two sets of N real numbers $X = \{x_1, x_2, ..., x_N\}$ and $\bar{X} = \{\bar{x_1}, \bar{x_2}, ..., \bar{x_N}\}$ (estimates). Let $A = \{a_1, ..., a_m\}$ and $\bar{A} = \{\bar{a_1}, ..., \bar{a_m}\}$ where if k is the aggregation granularity, $l_i = min(ik, N) - (i-1)k$, $m = \lceil \frac{N}{k} \rceil$, then $a_i = \sum_{j=(i-1)k+1}^{min(ik,N)} x_j$ and $\bar{a_i} = \sum_{j=(i-1)k+1}^{min(ik,N)} \bar{x_j}$. If $\epsilon_i = a_i - \bar{a_i}$ and it is known that $X \sim D$ where μ is the mean of the distribution D (i.e. $E[X]$) and σ^2 is the variance then,*

$$E[\sum_{j=1}^{N}(x_j - \bar{x_j})^2] \geq E[\sum_{j=1}^{N}(x_j - \hat{x_j})^2] \qquad (1)$$

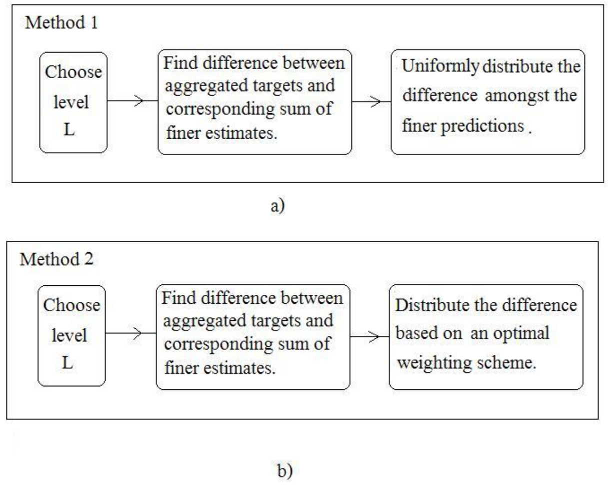
Method 1
Choose level L
Find difference between aggregated targets and corresponding sum of finer estimates.
Uniformly distribute the difference amongst the finer predictions.
a)
Method 2
Choose level L
Find difference between aggregated targets and corresponding sum of finer estimates.
Distribute the difference based on an optimal weighting scheme.
b)

Figure 4: The two methods represented in figure 2 are illustrated above. The first one evenly distributes the differences while the second one distributes the differences based on a convex weighting scheme.

where $\hat{x_j} = \bar{x_j} + \alpha_{\lceil \frac{j}{k} \rceil}^{(j \mod l_{\lceil \frac{j}{k} \rceil}+1)} \epsilon_{\lceil \frac{j}{k} \rceil}$, $\alpha_{\lceil \frac{j}{k} \rceil}^{(j \mod l_{\lceil \frac{j}{k} \rceil}+1)} \geq 0$ *and* $\sum_{i=1}^{l_p} \alpha_p^{(i)} = 1 \ \forall p \in \{1, ..., \lceil \frac{N}{k} \rceil\}$. *The optimal alphas that minimize the expectation on the right side of the inequality in equation 1 are given by,*

$$\alpha_p^{(i)} = \frac{1}{l_p \epsilon_p^2}[l_p \epsilon_p(\mu - \bar{x}_{i+k(p-1)}) - (2l_p - 1)(\sigma^2 + \mu^2)]; i \neq l_p$$

$$\alpha_p^{(l_p)} = \frac{1}{l_p \epsilon_p^2}[(2l_p - 1)(l_p - 1)(\sigma^2 + \mu^2) - l_p \epsilon_p((l_p - 1)\mu + \bar{x}_{i+k(p-1)} - a_p)]$$

A high level description of the methods formally described in lemma 1 and in lemma 3 are shown in figure 4.

2.2 Choosing between Multiple levels

Based on the proofs (in the appendix) of the results in the previous subsection, we observe that the reduction in MSE by applying our method is a function of the aggregation granularity and the accuracy of the estimates of the aggregated targets. In particular, the smaller the aggregation granularity and the lower the error the more significant the improvement. However, if we have estimates of the aggregated targets at multiple levels of granularity with varying accuracy, in the general case, it is not clear as to which level will lead to the most improvement. For example, at $k = 2$ we might have an error of 0.2 and at $k = 3$ we might have an error of 0.15. In this case, it is not clear whether to use the estimates of the aggregated target at level 2 or level 3. Note that if the error at level 3 was more than that at level 2 then the choice is obvious and we would choose level 2. Hence, we see that in choosing the appropriate level there is a trade-off between the aggregation granularity and the error of the estimates of the aggregated targets.

Criterion: If k is the aggregation granularity and MSE_k

denotes the mean squared error of the aggregated targets at aggregation granularity k then the level that is most likely to lead to maximum improvement in the predictions of the target is given by,

$$L = \min_{k} \operatorname*{argmin}_{k} kMSE_k \tag{2}$$

Hence, if there are multiple k values with the same value of the objective we choose the minimum k. If $L = 1$ is the answer then that means that the aggregated targets at coarser levels will most likely not help in improving the predictive accuracy.

One may ask what the thought process is in using the estimates of the aggregated targets if one is able to compute the MSE at that level which implies one knows true values of the aggregated targets. The point of this exercise is to come up with a criterion that in a traditional machine learning setting can be used to choose a regression model at a certain level of aggregation based on the training set followed by using this model to estimate aggregated targets on the test set, which then can be used to improve predictions of the target on the test set. An algorithm for the same is described below.

Algorithm for choosing level/model in traditional settings: In a standard machine learning setting where we have a training and a test set, we can train M models one for each of the M levels of aggregation on the training set and use the criterion mentioned in equation 2 to decide the best level and the corresponding model to be used to improve the predictions on the test set (i.e. of a model built at $k = 1$). If the test set size is N, potentially M could be N, however it makes little sense to build models beyond a certain level for mainly two reasons: 1) the corresponding dataset sizes (due to aggregation) at or beyond that level or aggregation granularity may be insufficient to train a model and 2) beyond a certain aggregation granularity even if we knew the exact values of the aggregated targets at those levels, the enhancement they produce in the quality of predictions is minuscule. As we will witness in the experimental section building models beyond $k = 10$ is quite unnecessary.

A flowchart describing the algorithm to decide the level and model is given in figure 5. As per the flowchart, we end up using the model R_L to predict aggregated targets on the test set. The estimates produced by R_L can then be used to improve the predictions of the model built to predict the target by using the strategy we outlined before. Note that the strategy changes depending on if in addition to these estimates we also use the estimates (or actuals if available) of the moments of the target distribution.

3. EXPERIMENTS

In this section we evaluate our proposed solutions on synthetic data as well as on 3 real industrial datasets obtained from diverse domains. In all the reported experiments we use ridge regression – which is a commonly used regression technique – as our baseline technique that will be used to predict the target as well as the aggregated targets (if not available). Moreover, whenever the algorithm in figure 5 is used, we build models till M is the test set size. However, we see that in each of the cases the optimal level L chosen by our algorithm is always < 10.

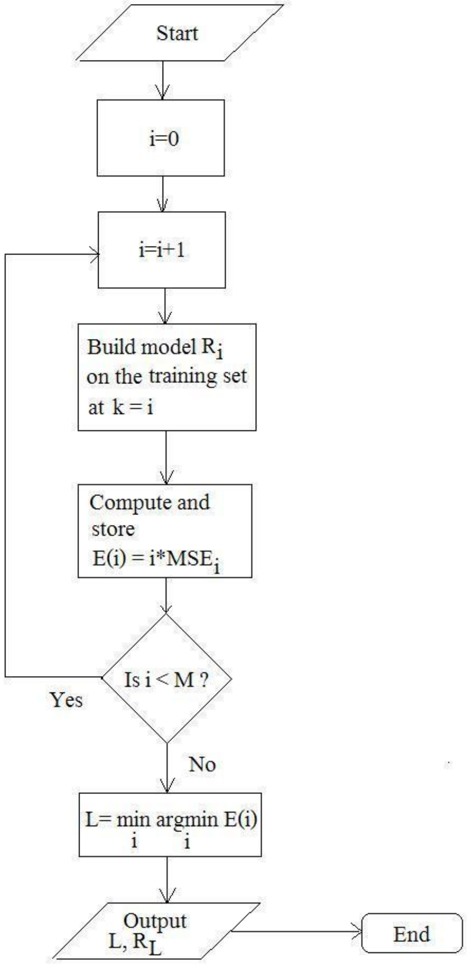
Start
i=0
i=i+1
Build model R_i on the training set at k = i
Compute and store E(i) = i*MSE_i
Is i < M ?
Yes
No
L= min argmin E(i)
i i
Output L, R_L
End

Figure 5: Algorithm for choosing the appropriate level and the corresponding regression function given that models are built at M levels i.e from $k = 1$ to $k = M$.

3.1 Synthetic Data Experiments

Setup: We generate synthetic datasets from an 11 dimensional Gaussian distribution, of which 10 are explanatory attributes and the last one is the target. We set the mean of this Gaussian to be the origin while the correlation matrix takes different values so as to generate different types of datasets. The general form of this correlation matrix is fixed however, in that all entries corresponding to correlation between two different explanatory attributes are set to zero – this is implied since we want independent explanatory attributes – while the standard deviation of all of the 11 attributes is set to σ and the correlation between the explanatory attributes and target is set to ρ. In order to generate datasets with different amounts of correlation between the explanatory attributes and target we vary ρ while to generate datasets with different variances of the individual attributes we vary σ. Moreover, to observe the behavior of our method on different dataset sizes, we create

$\sigma \downarrow$,$\rho \rightarrow$	0.8	0.6	0.4	0.2
0.8	0.15,0.27	0.09,0.23	0.11,0.33	0.31,0.42
0.6	0.19,0.20	**0.04**,0.13	**0.03**,0.33	0.52,0.47
0.4	0.16,0.22	**0.03**,0.12	**0.01**,0.16	0.31,0.45
0.2	0.27,0.41	0.18,0.32	0.37,0.36	0.48,0.61

Table 1: For dataset of size 100 we see above the $p - values$ of the paired t-test. The entries before the comma are $p - values$ for testing hypothesis H_0 and entries following the comma are $p - values$ for testing the hypothesis T_0. The entries in bold are cases where the corresponding null hypothesis are rejected.

$\sigma \downarrow$,$\rho \rightarrow$	0.8	0.6	0.4	0.2
0.8	0.23,0.26	0.08,0.09	0.10,0.27	0.24,0.39
0.6	0.16,0.13	**0.03**,**0.04**	**0.02**,**0.03**	0.43,0.46
0.4	0.17, 0.21	**0.03**,**0.02**	**0.02**,**0.01**	0.28,0.41
0.2	0.34,0.32	0.17,0.25	0.33,0.41	0.42,0.53

Table 2: For dataset of size 1000 we see above the $p - values$ of the paired t-test. The entries before the comma are $p - values$ for testing hypothesis H_0 and entries following the comma are $p - values$ for testing the hypothesis T_0. The entries in bold are cases where the corresponding null hypothesis are rejected.

datasets of size 100 and 1000 for each ρ and σ combination that we consider. In particular, we create datasets for $\rho = \{0.8, 0.6, 0.4, 0.2\} \times \sigma = \{0.8, 0.6, 0.4, 0.2\}$ as seen in tables 1 (dataset size = 100) and 2 (dataset size = 1000). For each of the combinations of these parameters we sample 100 datasets.

Each dataset that is obtained which is of a particular size and sampled with particular ρ and σ, is randomly split into train (70%) and test sets (30%). This is done 50 times to generate 50 different train and test splits on the same dataset. On each training dataset we first learn a model using just ridge regression (R), then we learn using ridge regression alongwith our algorithm in figure 5 (R+A) and finally we learn using ridge regression alongwith our algorithm and moment information (R+A+M) estimated from the training set which gives us an estimate of the optimal weights as per lemma 3. On the corresponding test sets we compute the MSE of each of these three techniques in predicting the target. Thus, for each combination of parameters we now get 50×100=5000 MSE values for the three techniques. The aggregates are formed using consecutive samples produced by the sampling process.

Using these MSE values we primarily want to decipher the parameter settings when R is worse than R+A and R+A is worse than R+A+M. To arrive at these conclusions in a statistically sound manner, we use the paired t-test [3] to confirm that the reductions in MSE from R to R+A and R+A to R+A+M are statistically significant. Hence, we carry out two hypothesis tests. The first one to see if the differences in MSE between R and R+A are significant and second one to see if the differences in MSE between R+A and R+A+M are significant. The null hypothesis in the first case is,

H_0 : The models R and R+A are equivalent.

The null hypothesis in the second case is,

T_0 : The models R+A and R+A+M are equivalent.

Observations: From the results in tables 1 and 2 we see that the hypothesis H_0 is rejected (i.e. $p - value < 0.05$) when the correlation between explanatory attributes and the target is neither too high nor too low and when the standard deviation of the attributes is moderate. A possible reason for this to happen is as follows: At high correlations the explanatory attributes are excellent predictors of the target and hence, a simple method is sufficient to be able to predict the target accurately. At low correlations the explanatory attributes contain very little information that can be used to predict the target irrespective of the modelling method used. At moderate correlations the explanatory attributes are reasonable predictors of the target. In this case however, when the variance of the target is high, aggregating using algorithm in figure 5 does not reduce the variance sufficiently until we have aggregated by more than a few levels and as mentioned before, higher the aggregation granularity lesser the improvement. When the variance of the target is low, further aggregating keeps the variance in the same regime which is low and hence the predictive power of the attributes does not change significantly at lower levels. At mid variances however, aggregation causes the variances to fall in the low regime and hence, at even low levels of aggregation we predict accurately which then enhances the performance of the model predicting the target.

From the results in tables 1 and 2 we see that the hypothesis T_0 is rejected when the dataset size is large and the other conditions are the same as that for hypothesis H_0 to be rejected. For R+A+M to work well it is required that R+A works at least reasonably since, if our estimates of the aggregated targets using algorithm in figure 5 are not accurate then irrespective of how we distribute these estimates the prediction quality will not improve by much. When R+A works really well there isn't much room for improvement. Moreover, a larger dataset size aids in obtaining more accurate estimates of the moments. Hence, the scenarios where R+A+M shows improvement over R+A is a subset of the cases where R+A shows improvement over R.

3.2 Real Data Experiments

We now test the efficacy of our approach on 3 real industrial datasets obtained from varied domains. Since, the records in each of these datasets are ordered by time, we respect this time ordering and use the first 70% of the records for training and the remaining 30% (i.e. most recent 30%) records for testing. We then report the MSE on the test set as a measure of evaluation for the different techniques with the results shown in figure 6.

Supply Chain Dataset: This dataset is obtained from an actual manufacturer and contains data at two levels namely; at the (finer) distribution center (DC) level and at the (coarser) manufacturer level. The goal is to predict the inventory position at a DC given past inventory positions and other attributes such as age of the inventory and product type (viz.

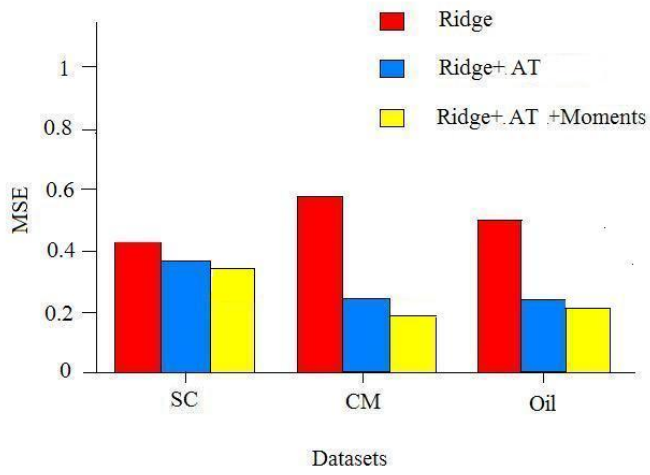

Figure 6: Comparison of three variants on three real datasets is shown above. SC stands for a Supply Chain dataset, CM stands for a chip manufacturing dataset and Oil stands for an Oil production dataset. AT implies aggregated target information.

egg beaters, pasta etc.). In addition, to this we also have information about the total amount shipped (aggregate information) from the manufacurer to meet the demands of the DC. Hence, in this case we do not need to use algorithm in figure 5 since, we already have aggregate information. We just need to use results in lemma 1 and lemma 3 (estimating the moments from the data). In our dataset there are 7 distribution centers and the data was collected daily for about a year (dataset size is 357).

Chip Manufacturing Dataset: In the chip manufacturing industry predicting speed of the wafers (collections of chips) accurately ahead of time can be crucial in choosing the appropriate set of wafers to send forward for further processing. Eliminating faulty wafers can save the industry a huge amount of resources in terms of time and money.

This dataset has 175 features where, the wafer speed is one of them. The other features are a combination of physical measurements and electrical measurements made on the wafer. The dataset size is 2361. In this case, aggregate information is unavailable and hence, we use the algorithm in figure 5 to estimate the aggregated targets, which can be viewed as estimating drift in the time series.

Oil Production Dataset: Oil companies periodically launch production logging campaigns to get an idea of the overall performance as well as to assess their individual performance at particular oil wells and reservoirs. These campaigns are usually expensive and laden with danger for the people involved in the campaign. Automated monitoring of oil production equipment is an efficient, risk free and economical alternative to the above solution.

The dataset we perform experiments on, is obtained from a major oil corporation. There are a total of 9 attributes in the dataset. These attributes are obtained from the sensors of a 3-stage separator which separates oil, water and gas. The 9 attributes are composed of 2 measured levels of the oil water interface at each of the 3 stages and 3 overall attributes. Our target is Daily production which indicates the amount of oil produced every day at the well. The dataset size is 992. In this case too, aggregate information is unavailable and hence, we use the algorithm in figure 5 to estimate the aggregated targets.

Observations: In figure 6, we observe the behavior of the three variants, namely: 1) ridge regression (R), 2) ridge regression using (actual or estimated) aggregated targets (R+A) and 3) ridge regression using (actual or estimated) aggregated targets alongwith estimated moments (R+A+M), on the three datasets. First we see that the performance improves consistently as we go from R to R+A and from R+A to R+A+M. However, the extent of the improvement differs in the 3 cases. A possible reason for the improvement from R+A to R+A+M being more significant on CM than on the other datasets could be that the larger dataset size leads to more accurate estimates of the moments as compared to the other datasets. The improvement from R to R+A is more pronounced on CM and Oil than on SC since, the L returned by algorithm in figure 5 is much lower than 7 – which is the aggregation granularity for SC – and the inaccuracies in the estimates of the aggregated targets for these two cases are only slight.

4. DISCUSSION

In this paper, we proposed a provable way of improving prediction quality with the help of accurate aggregate or coarser information. In cases where we may have (estimates of) aggregated targets at multiple levels we provided a way of choosing the optimal level so as to maximize the improvement in prediction quality. We have provided an algorithm for the same, in standard machine learning settings where aggregate information may not be available from an independent source. Moreover, using estimates of the moments of the target distribution, we have provided a way of better distributing the aggregate information so as to further enhance the predictive accuracy.

In the future, it may be desirable to choose multiple levels rather than just a single level and in some provable manner use the corresponding aggregated targets in an attempt to obtain results superior to the ones in this paper. Moreover, when the data is unordered one could first cluster the data and then apply the suggested algorithms to further enhance the predictive power. A method that encapsulates these ideas might be of some interest. In any case, we believe that we have laid the basic groundwork for the creation of advanced methods such as these.

Acknowledgement

I would like to thank Jayant Kalagnanam for providing data and exposing me to such problems. I would like to thank Naoki Abe and Aurelie Lozano for helpful discussions. I would also like to thank Brian White and Katherine Turner for proofreading the paper.

APPENDIX

Proof of Lemma 1

PROOF. Consider two sets of N real numbers $X = \{x_1, x_2, ..., x_N\}$ and $\bar{X} = \{\bar{x_1}, \bar{x_2}, ..., \bar{x_N}\}$ (estimates). Let $A = \{a_1, ..., a_m\}$

and $\bar{A} = \{\bar{a_1}, ..., \bar{a_m}\}$ where if k is the aggregation granularity, $l_i = min(ik, N) - (i-1)k - 1$, $m = \lceil \frac{N}{k} \rceil$, then $a_i = \sum_{j=(i-1)k+1}^{min(ik,N)} x_j$ and $\bar{a_i} = \sum_{j=(i-1)k+1}^{min(ik,N)} \bar{x_j}$. Let $\epsilon_i = a_i - \bar{a_i}$ and $\hat{x_j} = \bar{x_j} + \frac{\epsilon_{\lceil \frac{j}{k} \rceil}}{l_{\lceil \frac{j}{k} \rceil}}$.

Let the mean squared error based on the original estimates i.e. $(\bar{x_i})$ be given by,

$$MSE_{old} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x_i})^2 \tag{3}$$

Hence, the mean squared error based on new estimates i.e. $(\hat{x_i})$ is given by,

$$\begin{aligned} &MSE_{new} \\ &= \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x_i})^2 = \frac{1}{N}\sum_{i=1}^{N}((x_i - \bar{x_i}) - \frac{\epsilon_{\lceil \frac{i}{k} \rceil}}{l_{\lceil \frac{i}{k} \rceil}})^2 \\ &= \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x_i})^2 - \frac{1}{N}[\sum_{i=1}^{N}(\frac{2}{l_{\lceil \frac{i}{k} \rceil}}(x_i - \bar{x_i})\epsilon_{\lceil \frac{i}{k} \rceil} - \frac{1}{l^2_{\lceil \frac{i}{k} \rceil}}(\epsilon_{\lceil \frac{i}{k} \rceil})^2)] \\ &= MSE_{old} - \frac{1}{N}A \end{aligned} \tag{4}$$

where $A = \sum_{i=1}^{N}(\frac{2}{l_{\lceil \frac{i}{k} \rceil}}(x_i - \bar{x_i})\epsilon_{\lceil \frac{i}{k} \rceil} - \frac{1}{l^2_{\lceil \frac{i}{k} \rceil}}(\epsilon_{\lceil \frac{i}{k} \rceil})^2)$. Now to prove our result we have to show that $A \geq 0$.

$$\begin{aligned} A &= \sum_{i=1}^{N}(\frac{2}{l_{\lceil \frac{i}{k} \rceil}}(x_i - \bar{x_i})\epsilon_{\lceil \frac{i}{k} \rceil} - \frac{1}{l^2_{\lceil \frac{i}{k} \rceil}}(\epsilon_{\lceil \frac{i}{k} \rceil})^2) \\ &= \sum_{p=1}^{\lceil \frac{N}{k} \rceil}(\frac{2}{l_p}(\sum_{i=(p-1)k+1}^{min(pk,N)} x_i - \sum_{i=(p-1)k+1}^{min(pk,N)} \bar{x_i})\epsilon_p - \frac{1}{l_p^2} l_p \epsilon_p^2) \\ &= \sum_{p=1}^{\lceil \frac{N}{k} \rceil}(\frac{2}{l_p}\epsilon_p^2 - \frac{1}{l_p}\epsilon_p^2) = \sum_{p=1}^{\lceil \frac{N}{k} \rceil}\frac{1}{l_p}\epsilon_p^2 \geq 0 \end{aligned} \tag{5}$$

Proof of Theorem 1

PROOF. The proof of the theorem follows from the proof of lemma 1 where we substitute ϵ_i with $\delta_i \in [0, 2\epsilon_i]$ in equation 4. With this we have,

$$MSE_{new} = MSE_{old} - \frac{1}{N}B \tag{6}$$

where $B = \sum_{i=1}^{N}(\frac{2}{l_{\lceil \frac{i}{k} \rceil}}(x_i - \bar{x_i})\delta_{\lceil \frac{i}{k} \rceil} - \frac{1}{l^2_{\lceil \frac{i}{k} \rceil}}(\delta_{\lceil \frac{i}{k} \rceil})^2)$. Now to prove our result we have to show that $B \geq 0$.

$$\begin{aligned} B &= \sum_{i=1}^{N}(\frac{2}{l_{\lceil \frac{i}{k} \rceil}}(x_i - \bar{x_i})\delta_{\lceil \frac{i}{k} \rceil} - \frac{1}{l^2_{\lceil \frac{i}{k} \rceil}}(\delta_{\lceil \frac{i}{k} \rceil})^2) \\ &= \sum_{p=1}^{\lceil \frac{N}{k} \rceil}(\frac{2}{l_p}(\sum_{i=(p-1)k+1}^{min(pk,N)} x_i - \sum_{i=(p-1)k+1}^{min(pk,N)} \bar{x_i})\delta_p - \frac{1}{l_p^2} l_p \delta_p^2) \\ &= \sum_{p=1}^{\lceil \frac{N}{k} \rceil}(\frac{2}{l_p}\delta_p\epsilon_p - \frac{1}{l_p}\delta_p^2) = \sum_{p=1}^{\lceil \frac{N}{k} \rceil}\frac{-1}{l_p}\delta_p(\delta_p - 2\epsilon_p) \end{aligned} \tag{7}$$

The above quadratic equation has 2 roots $\delta_p = 0$ and $\delta_p = 2\epsilon_p$ and we already know that $B \geq 0$ when $\delta_p = \epsilon_p$. Since, $\epsilon_p \in [0, 2\epsilon_p]$ and the function is a quadratic in δ_p we have $B \geq 0$ $\forall \delta_p \in [0, 2\epsilon_p]$.

Proof of Lemma 2

PROOF. In equation 5 substituting the alphas we have,

$$A = \sum_{p=1}^{\lceil \frac{N}{k} \rceil} 2\epsilon_p \sum_{i=(p-1)k+1}^{min(pk,N)} [(x_i - \bar{x_i})\alpha_i - \frac{\epsilon_p}{2}\alpha_i^2] \tag{8}$$

We thus have to show that when all alphas for a particular p are not equal then there always exist X and $\bar{X}$ such that $A < 0$. We can show this by proving that there always exist $\{x_{(p-1)k+1}, ..., x_{min(pk,N)}\}$ and $\{x_{(p-\bar{1})k+1}, ..., x_{min\bar{(}pk,N)}\}$ such that the above equation for any particular p is less than zero and hence, if we replicate this case for all p then their sum is less than zero which implies $A < 0$. With this we have to show that for any p (in our setting), $2\epsilon_p \sum_{i=(p-1)k+1}^{min(pk,N)}[(x_i - \bar{x_i})\alpha_i - \frac{\epsilon_p}{2}\alpha_i^2] \leq 0$.

Without loss of generality (w.l.o.g.) we will prove the above result for $p = 1$ and the proof should be valid for all p. Hence, we will show that when all alphas for $p = 1$ are not equal then there always exist $\{x_1, ..., x_k\}$ and $\{\bar{x_1}, ..., \bar{x_k}\}$ such that, $2\epsilon_1 \sum_{i=1}^{k}[(x_i - \bar{x_i})\alpha_i - \frac{\epsilon_1}{2}\alpha_i^2] \leq 0$.

Since all alphas are not equal, w.l.o.g. assume that $\alpha_1 > \alpha_2$ where $\alpha_1 \geq \alpha_i$ $\forall i \in \{1, ..., k\}$ and $\alpha_2 \leq \alpha_i$ $\forall i \in \{1, ..., k\}$. We will prove the result by dividing it into 2 cases. Case 1 is $\epsilon_1 \geq 0$ and case 2 is $\epsilon_1 \leq 0$. Notice that we have freedom to choose values for X and $\bar{X}$ to prove our result.

Case 1: We choose x_i and $\bar{x_i}$ such that $x_i = \bar{x_i}$ $\forall i \in \{3, ..., k\}$ and $x_2 - \bar{x_2} \geq \bar{x_1} - x_1 \geq 0$. This forces $\epsilon_1 \geq 0$ as desired. Hence, for the previous equation to be true, a sufficient condition is, $\alpha_1(x_1 - \bar{x_1}) + \alpha_2(x_2 - \bar{x_2}) \leq 0$ which implies $\bar{x_1} - x_1 \geq \frac{\alpha_2}{\alpha_1}(x_2 - \bar{x_2})$. We can always find $x_1, \bar{x_1}, x_2$ and $\bar{x_2}$ such that $x_2 - \bar{x_2} \geq \bar{x_1} - x_1 \geq \frac{\alpha_2}{\alpha_1}(x_2 - \bar{x_2}) \geq 0$ $\forall \alpha_i$ where $i \in \{1, ..., k\}$.

Case 2: This is analogous to case 1. All the inequalities in case 1 can be reversed and hence, we need to find $x_1, \bar{x_1}, x_2$ and $\bar{x_2}$ such that $x_2 - \bar{x_2} \leq \bar{x_1} - x_1 \leq \frac{\alpha_2}{\alpha_1}(x_2 - \bar{x_2}) \leq 0$ $\forall \alpha_i$ where $i \in \{1, ..., k\}$, which is definitely possible.

Proof of Lemma 3

PROOF. Since we take expectations with respect to the underlying distribution for the result of this lemma the objective we have to maximize to get the optimal alphas is the expected value of equation 8 given a_p and $\bar{a_p}$, i.e. $E[A|a_p, \bar{a_p}]$. This function is concave in the alphas and hence by forming the lagrangian and maximizing the objective given the constraints on the alphas we get, $\alpha_p^{(i)} = \frac{1}{2\epsilon_p^2}[2\epsilon_p(\mu - \bar{x}_{i+k(p-1)} + \lambda)]; i \neq l_p$ and $\lambda = \frac{-2}{l_p}(2l_p - 1)(\sigma^2 + \mu^2)$, where λ is the lagrange parameter. Notice that $\alpha_p^{(l_p)}$ is uniquely defined since the alphas sum to 1. With this the optimal alphas are given by, $\alpha_p^{(i)} = \frac{1}{l_p\epsilon_p^2}[l_p\epsilon_p(\mu - \bar{x}_{i+k(p-1)}) - (2l_p - 1)(\sigma^2 + \mu^2)]; i \neq l_p$ and $\alpha_p^{(l_p)} = \frac{1}{l_p\epsilon_p^2}[(2l_p - 1)(l_p - 1)(\sigma^2 + \mu^2) - l_p\epsilon_p((l_p - 1)\mu + \bar{x}_{i+k(p-1)} - a_p)]$

A. REFERENCES

[1] A. Arnold, Y. Liu, and N. Abe. Temporal causal

modeling with graphical granger methods. In *KDD*. ACM, 2007.

[2] A. Dhurandhar. Multistep time series prediction in complex instrumented domains. In *Large-scale Analytics for Complex Instrumented Systems workshop, ICDM*. IEEE, 2010.

[3] T. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10:1895–1923, 1998.

[4] F. Fleuret and D. Geman. Coarse-to-fine face detection. *Int. J. Comput. Vision*, 41:85–107, 2001.

[5] C. Jackson, N. Best, and S. Richardson. Hierarchical related regression for combining aggregate and individual data in studies of socio-economic disease risk factors. *Journal Of The Royal Statistical Society Series A*, 171(1):159–178, 2008.

[6] Y. Liu, J. Kalagnanam, and O. Johnsen. Learning dynamic temporal graphs for oil-production equipment monitoring system. In *KDD*, pages 1225–1234. ACM, 2009.

[7] D. Munoz, J. Bagnell, and M. Hebert. Stacked hierarchical labeling. In *ECCV*, 2010.

[8] M. Park, T. Hastie, and R. Tibshirani. Averaged gene expressions for regression. *Biostatistics*, pages 212–227, 2007.

[9] B. Sapp, A. Toshev, and B. Taskar. Cascaded models for articulated pose estimation. In *ECCV*, 2010.

[10] P. Slav. Coarse-to-fine natural language processing. Phd Thesis UC Berkeley, 2009.

[11] D. Weiss and B. Taskar. Structured prediction cascades. In *Proc. AISTATS*, 2010.