Cross Domain Classification

- To link two domains together, a relationship between the two domains needs to be found.

- In a classification task, the correspondence between the classes in the two domains may be a good relationship.

- This may be illustrated by the simplest case (**shared data points**) where two domains contain the same set of data points but classifies them from different aspects.

- Then more complicated cases (**common feature, link to topic representation**) will be considered.

Cross Domain Classification - Shared Data Points

- Assume two eBay shops sell identical products (d1, d2, …) but with different classification (class1, class2, etc. for shop 1 and Class1, Class2 etc. for shop 2)

- The probability of a product to be in class i and Class j can be defined as:

$$\frac{The\ number\ of\ products\ both\ in\ class\ i\ and\ Class\ j}{The\ total\ number\ of\ products\ in\ class\ i\ or\ Class\ j} = \frac{class\ i \cap Class\ j}{class\ i \cup Class\ j}$$

- Then, a graphical model as shown in next slides can be built

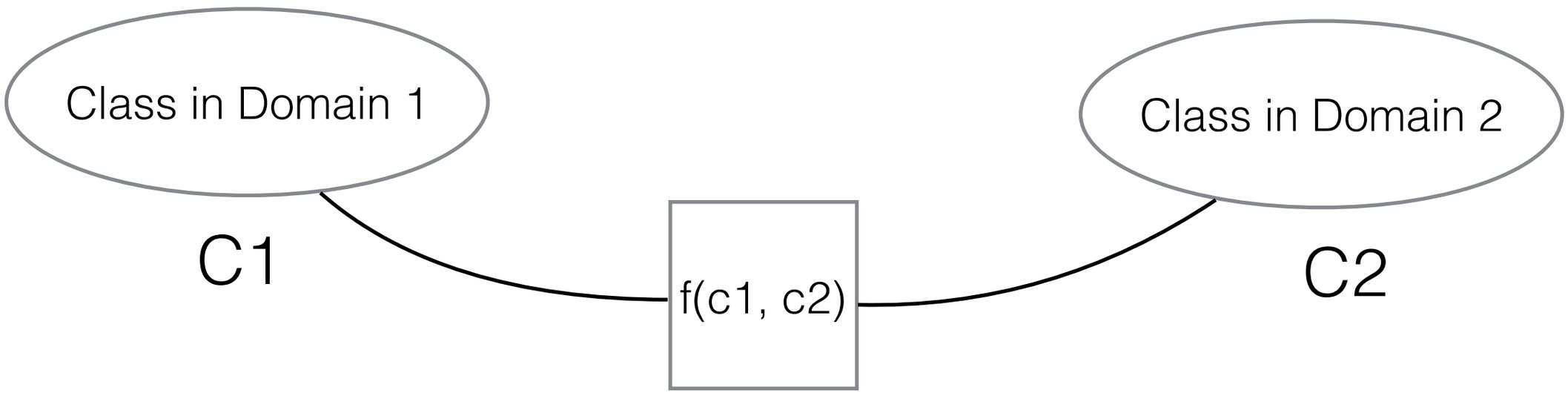
Cross Domain Classification - Shared Data Points

Output of classifier 1

class 1	class 2	…
probability 1	probability 2	….

Output of classifier 2

Class 1	Class 2	…
Probability 1	Probability 2	….



- Maximize P(class) when classifying domain 1
- Similar trick can be applied in indirect relations, except class 1 in the table contains all the data points that is classified to class 1 by classifier 1
- Or we may apply to EM algorithm to estimate the CPD

	class 1	class 2	…
Class 1	$\frac{class1 \cap Class1}{class1 \cup Class1}$	$\frac{class2 \cap Class1}{class2 \cup Class1}$	…
Class 2	$\frac{class1 \cap Class2}{class1 \cup Class2}$	$\frac{class2 \cap Class2}{class2 \cup Class2}$	…
…	…	…	…

Cross Domain Classification - Common Feature

- The principle can be extended to more complicated cases where no data is shared but all the data points in two domains are represented with common feature (say, texts)

- The key idea is that classify d1 from domain 1 with Classifier in Domain 2, and use the output as the Class of d1 in Domain 2 and vice versa.

- Then every data point has a class in both domain and it is like the previous case.

- But here is a problem: will a CPD constructed with the two classifiers really help the classification of the two classifiers? — Nothing new is here.

Cross Domain Classification - Common Feature

- If the CPD constructed with the two classifiers does not work well, we may use EM method to improve the CPD:

Initialize the CPD with the output of the two classifiers
loop until converge:
E step:
Use the overall classifier to classify all the data points and assign their classes in the other domain
M step:
Use the classes in both domain of each product to construct the CPD and update the overall classifier.

Cross Domain Classification - Confusion

- There should be other methods to update the CPD. Especially a way to improve training accuracy directly.
- Not very sure the behavior of the EM method without any test

Cross Domain Classification - Class Topic Consistency

- A better way to estimate the CPD may be to utilize the topic composition of each classes.

- It is natural to assume that each classes in both domain can be characterized with finer topics. And the classes have more topics in common may have more common products.

- If we do a LDA on all the texts within a class and character the class with the output topics, it is reasonable to assume that the overlap of topics of each classes should be consistent with the CPD

Cross Domain Classification - Class Topic Consistency

- It is also reasonable to characterize each class as a list of mixing proportions
- Then the CPD should be consistent with the similarity between the mixing proportions of two classes

Too far to know for sure.

But the idea is that topic representation and the CPD should somehow be consistent — they both assumes common feature

Major difficulties

- The requirement is too strict — common data sets or common feature — can we relax it?

- Too many design choices — we may choose to use one CPD as shown in slide 2; or two CPDs one from domain 1 to domain 2, the other one from domain 2 to domain 1

- How to start? Is there a proper data set? Can we validate our assumptions step by step?

- Too task specific, can we generate to other tasks rather than classification? What are the other tasks that is of interest?