Frustratingly Easy Domain Adaptation

Daumé III, H. 2007.

Kang Ji

Language Processing for Different Domains and Genres

WS 2009/10

Overview

- Motivation
- Annotation
- Core Approach
  - Prior Works
  - Feature Annotation
  - Kernelized Version
- Some Experimental Results

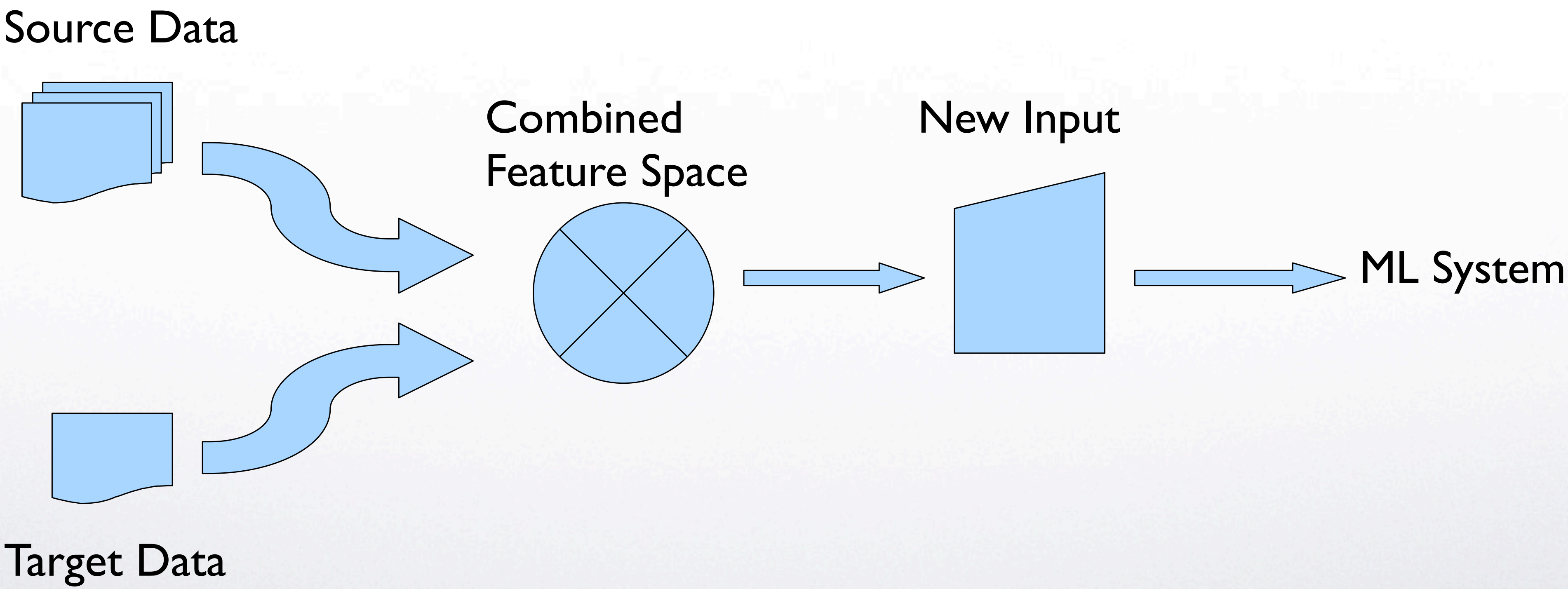
A common special case

- Suppose we have a NLP system focusing on news document, and now want to migrate it into biographic domain

  Would there be any difference if we

  - have quite some biographic documents(target data) and lots of news documents.
  - only have news documents(source data).

Rough Idea


Source Data
Combined
Feature Space
New Input
ML System
Target Data

ML approaches

- Now we simplified the task to a standard machine learning problem
  - Fully supervised learning: annotated corpus
  - Semi-supervised learning: large unannotated corpus, annotated corpus from the later target data

Some Annotations

- Input space X,
- Output space Y,
- Samples: D^s D^t

D^s is a collection of N examples and D^t is a collection of M examples (where, typically, $N \gg M$).

Some Annotations

- Distribution on the source and target domains: $\mathcal{D}^s$ $\mathcal{D}^t$

- learning function $h : \mathcal{X} \rightarrow \mathcal{Y}$

  $\mathcal{X} = R^F$ and that $\mathcal{Y} = \{-1,+1\}$

Prior works

- The SRCONLY baseline ignores the target data and trains a single model, only on the source data.
- The TGTONLY baseline trains a single model only on the target data.
- The ALL baseline simply trains a standard learning algorithm on the union of the two datasets.

Prior works

- The WEIGHTED baseline: re-weight examples from D^s.

  in case that $N \gg M$, so if $N = a \times M$, we may weight each example from the source domain by $1/a$.

Prior works

- The PRED baseline is based on the idea of using the output of the source classifier as a feature in the target classifier.
- The LININT baseline, we linearly interpolate the predictions of the SRCONLY and the TGTONLY models.

Prior works

- The PRIOR model is to use the SRCONLY model as a prior on the weights for a second model, trained on the target data.
- The maximum entropy classifiers model by Daum´e III and Marcu (2006), learns three models and justifies on a per-example basis.

Feature Augmentation

- $\Phi^s, \Phi^t : X \rightarrow \dot{X}$ mapping for source and target data respectively, then define $\dot{X} = R^{3F}$, we get
- $\Phi^s(x) = \langle x,x,0 \rangle$; $\Phi^t(x) = \langle x,0,x \rangle$
- the features which are made into three: general version, source-specific version, target-specific version
- get some ideas? examples coming---> black board

a simple and pleasing result

- $\check{K}(x, x') = 2K(x, x')$ same domain
- $\check{K}(x, x') = K(x, x')$ diff. domain
  - the data point from the target domain has ***twice*** as much influence as the data point from source domain on the prediction of the test target data.

Extension to Multi-domain adaption

- For a K-domain problem, we simply expand the feature space from R^{3F} to $R^{(K+1)F}$
- "+1" stands for the "general domain"

Why better

- This model optimize the feature weights jointly, thus there's no need to cross-validate to estimate good hyperparameters for each task as the PRIOR model does.
- Also it means that the single supervised learning algorithm that is run is allowed to regulate the trade-off between source/target and general weights.

Task Statistics

- Table 1: Task statistics;
- columns are task, domain,size of the training, development and test sets, and the number of unique features in the training set.
- Feature sets: lexical information (words,stems, capitalization, prefixes and suffixes), membership on gazetteers, etc.

Task	Dom	# Tr	# De	# Te	# Ft
	bn	52,998	6,625	6,626	80k
	bc	38,073	4,759	4,761	109k
ACE-	nw	44,364	5,546	5,547	113k
NER	wl	35,883	4,485	4,487	109k
	un	35,083	4,385	4,387	96k
	cts	39,677	4,960	4,961	54k
CoNLL-	src	256,145	-	-	368k
NER	tgt	29,791	5,258	8,806	88k
PubMed-	src	950,028	-	-	571k
POS	tgt	11,264	1,987	14,554	39k
CNN-	src	2,000,000	-	-	368k
Recap	tgt	39,684	7,003	8,075	88k
	wsj	191,209	29,455	38,440	94k
	swbd3	45,282	5,596	41,840	55k
	br-cf	58,201	8,307	7,607	144k
Tree	br-cg	67,429	9,444	6,897	149k
bank-	br-ck	51,379	6,061	9,451	121k
Chunk	br-cl	47,382	5,101	5,880	95k
	br-cm	11,696	1,324	1,594	51k
	br-cn	56,057	6,751	7,847	115k
	br-cp	55,318	7,477	5,977	112k
	br-cr	16,742	2,522	2,712	65k

Task	Dom	SrcOnly	TgtOnly	All	Weight	Pred	LinInt	Prior	Augment	T<S	Win
ACE-NER	bn	4.98	2.37	2.29	2.23	2.11	2.21	2.06	**1.98**	+	+
	bc	4.54	4.07	3.55	3.53	3.89	4.01	**3.47**	**3.47**	+	+
	nw	4.78	3.71	3.86	3.65	3.56	3.79	3.68	**3.39**	+	+
	wl	2.45	2.45	**2.12**	**2.12**	2.45	2.33	2.41	**2.12**	=	+
	un	3.67	2.46	2.48	2.40	2.18	2.10	2.03	**1.91**	+	+
	cts	2.08	0.46	0.40	0.40	0.46	0.44	**0.34**	**0.32**	+	+
CoNLL	tgt	2.49	2.95	1.80	**1.75**	2.13	**1.77**	1.89	**1.76**		+
PubMed	tgt	12.02	4.15	5.43	4.15	4.14	3.95	3.99	**3.61**	+	+
CNN	tgt	10.29	3.82	3.67	3.45	3.46	3.44	**3.35**	**3.37**	+	+
Tree bank-Chunk	wsj	6.63	4.35	4.33	4.30	4.32	4.32	4.27	**4.11**	+	+
	swbd3	15.90	4.15	4.50	4.10	4.13	4.09	3.60	**3.51**	+	+
	br-cf	5.16	6.27	4.85	4.80	4.78	**4.72**	5.22	5.15		
	br-cg	4.32	5.36	**4.16**	**4.15**	4.27	4.30	4.25	4.90		
	br-ck	5.05	6.32	5.05	4.98	**5.01**	**5.05**	5.27	5.41		
	br-cl	5.66	6.60	5.42	**5.39**	**5.39**	5.53	5.99	5.73		
	br-cm	3.57	6.59	**3.14**	**3.11**	3.15	3.31	4.08	4.89		
	br-cn	4.60	5.56	4.27	4.22	**4.20**	**4.19**	4.48	4.42		
	br-cp	4.82	5.62	4.63	**4.57**	**4.55**	**4.55**	4.87	4.78		
	br-cr	5.78	9.13	5.71	5.19	5.20	**5.15**	6.71	6.30		
Treebank-brown		6.35	5.75	4.80	4.75	4.81	4.72	4.72	**4.65**	+	+

Table 2: Task results.

Task results

Model Introspection

- "broadcast news" contains no capitalization
- "broadcast conversation"
- "newswire"
- "Weblog"
- "usenet" may contain many email addresses and URLs
- "conversational telephone speech"

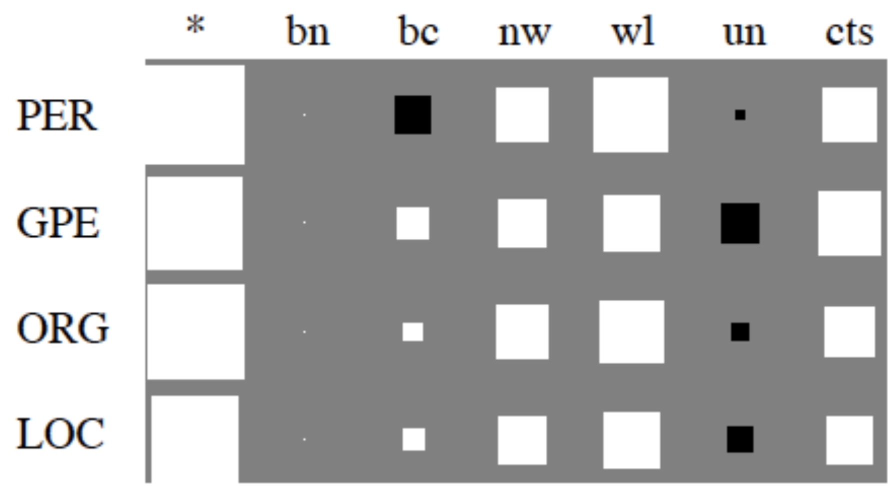


Figure 1: Hinton diagram for feature /Aa+/ at current position.

Implementation Demo

- http://public.me.com/jikang/easyadapt.pl.zip (only 10 line perl script, how elegant!)

Reference

- Hal Daum´e III, 2007. Frustratingly Easy Domain Adaptation
- Hal Daume III,Daniel Marcu,2006. Domain Adaptation for Statistical Classifiers