Using Multiple Sources to Construct a Sentiment Sensitive Thesaurus for Cross-Domain Sentiment Classification

Danushka Bollegala, David Weir, John Carroll

THE UNIVERSITY OF TOKYO

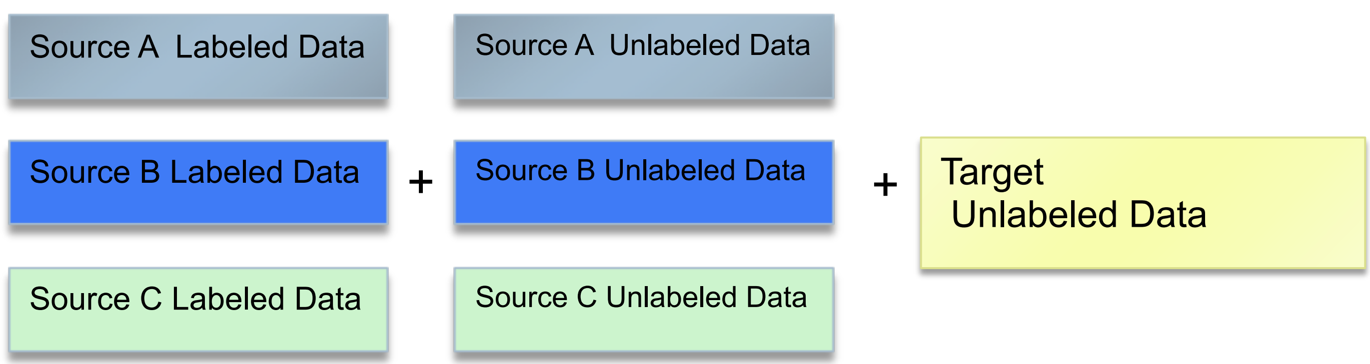
University of Sussex

Cross-Domain Sentiment Classification

- In this talk, we focus on document-level, binary sentiment classification.
- Learn to classify the sentiment (positive/negative) about a particular product types (*source domains*) and use the learnt classifier to classify sentiment on a different product type (*target domain*).
- Why would we want to do that?
  - There are millions of product types in the market and new products are continuously emerging.
  - However, we do not have sufficient labeled data for all product types to train a supervised sentiment classifier.
  - Manually labeling data for each new domain is costly.

Problem Definition

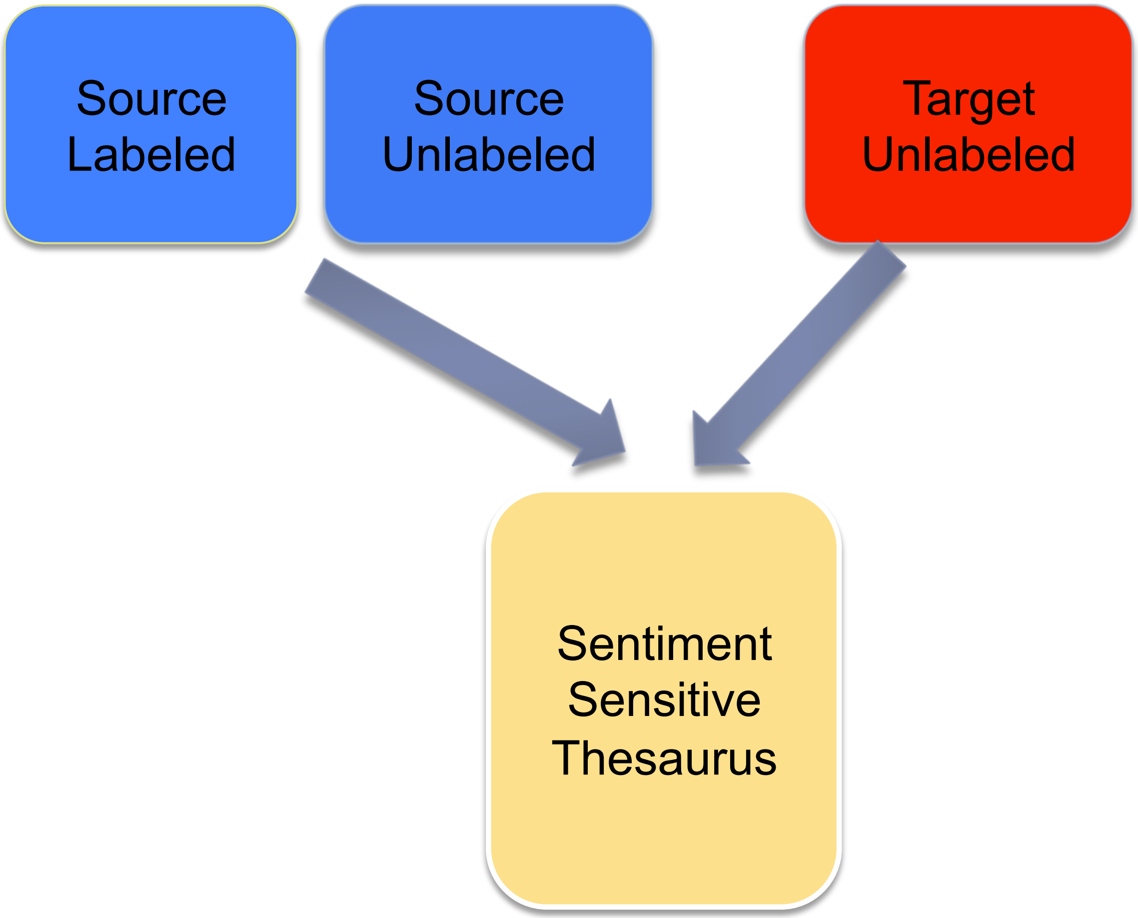
- *Unsupervised* Cross-Domain Sentiment Classification
  - Given labeled data (+/- labeled reviews) for the source domains and unlabeled data (reviews with no sentiment labels) for both source and target domains, learn a binary classifier **h** to classify the sentiment on target domain reviews.
- Other variants of Domain Adaptation (not considered in this work)
  - supervised domain adaptation (Daume, ACL 2007)
    - use some labeled data for the target domain as well.
  - semi-supervised domain adaptation (Daume, NIPS, 2010)
    - use both labeled and unlabeled data for the target domain



Training dataset

Challenges and the Proposed Solution

- Feature Mismatch
  - Features that indicate a particular sentiment in the source domain might not be relevant for the target domain.
  - interesting book ≈ energy saving grill
  - boring movie ≈ rusty knives
- Which source domain features are *related* to which target domain features?
  - We create a *sentiment sensitive thesaurus* to group related source and target domain features
  - We propose a method to *expand* feature vectors at train and test times for a binary sentiment classifier.



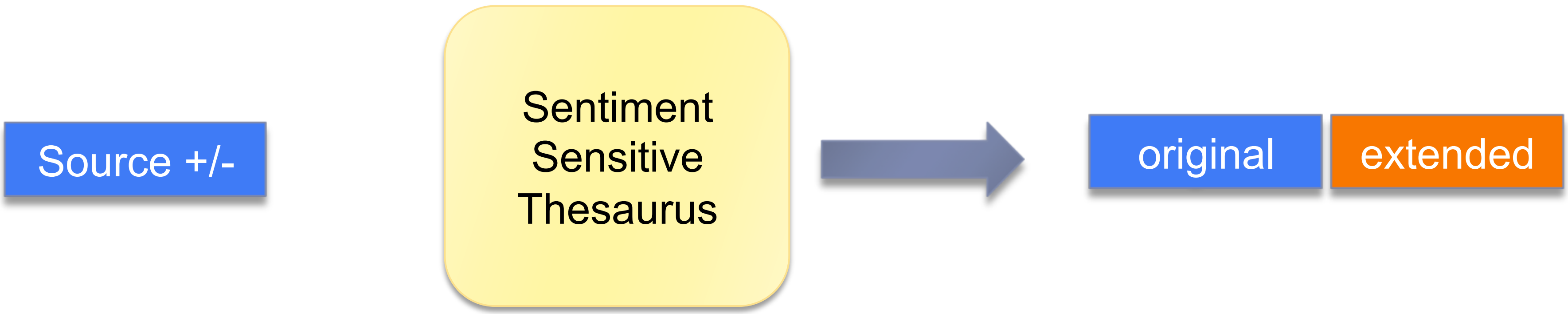
Source Labeled
Source Unlabeled
Target Unlabeled
Sentiment Sensitive Thesaurus

Training

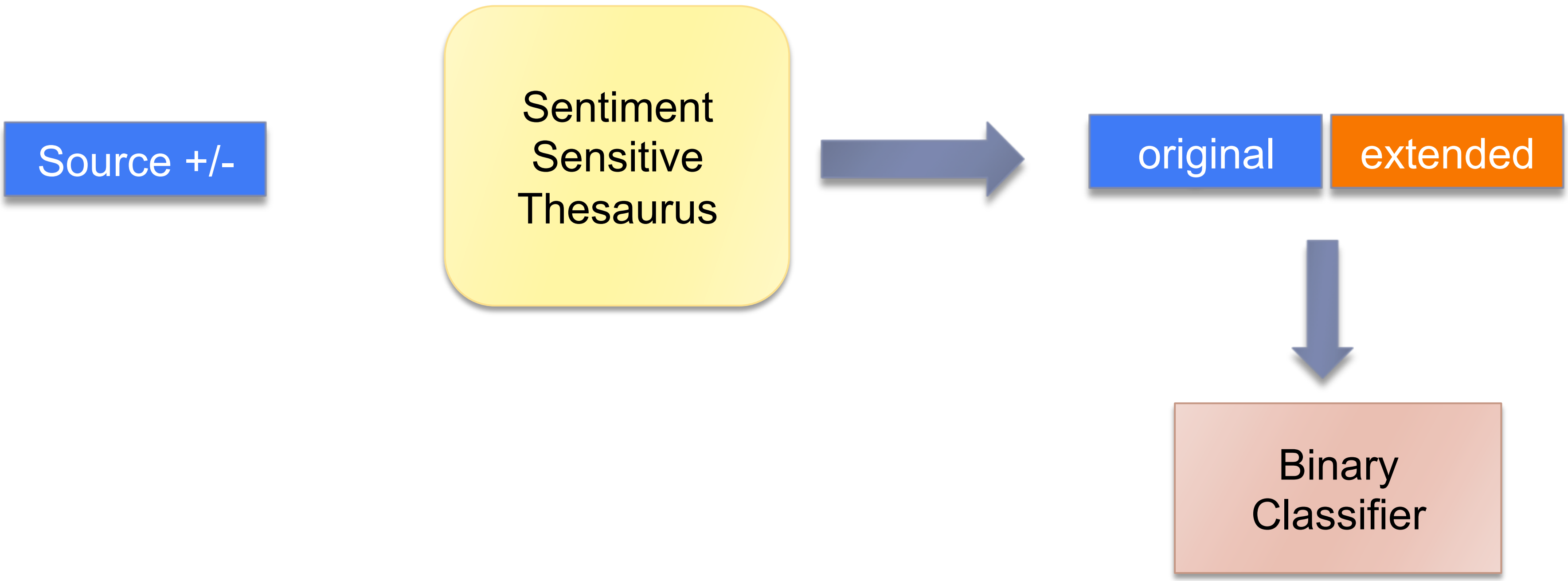
Source Labeled

Sentiment Sensitive Thesaurus

Training



Training

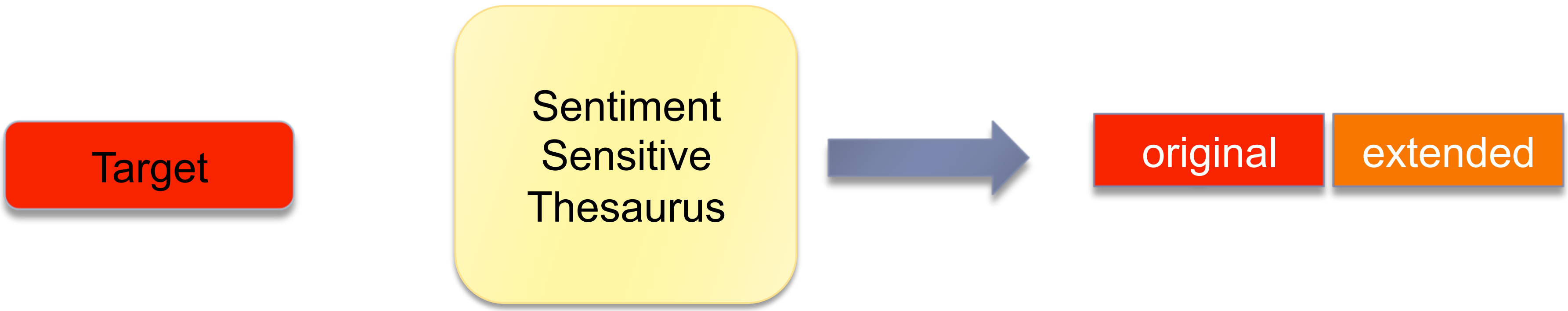

Source +/-
Sentiment Sensitive Thesaurus
original
extended
Binary Classifier

Testing

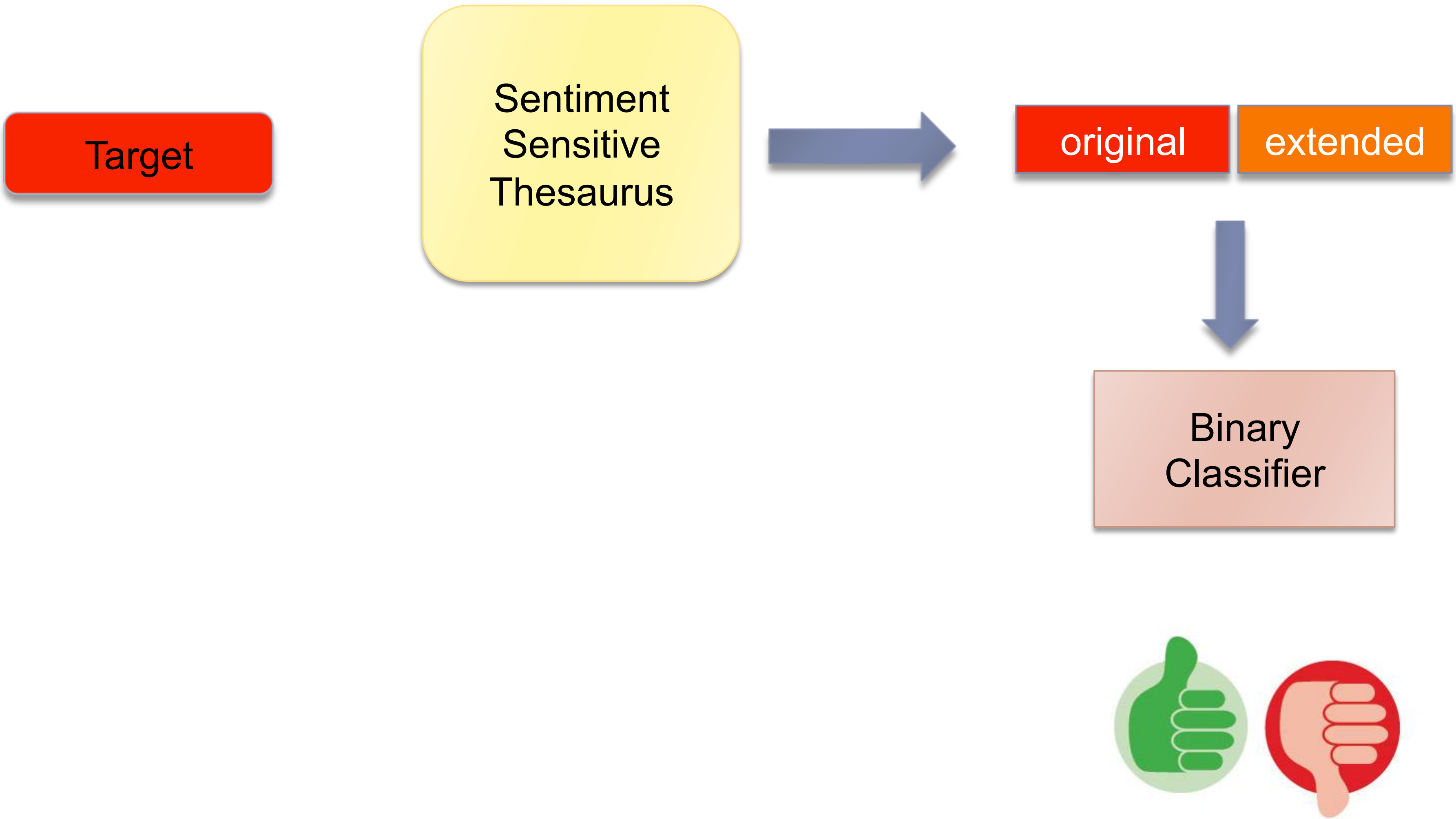
Target

Sentiment Sensitive Thesaurus

Testing



Testing


Target
Sentiment Sensitive Thesaurus
original
extended
Binary Classifier

Creating a Sentiment Sensitive Thesaurus (1/3)

Excellent and broad survey of the development of civilization
Excellent/JJ and/CC broad/JJ survey/NN1 of/IO the/AT development/NN1 of/IO civilization/NN1

- unigrams and bigrams of words in a review are called **lexical elements** and act as the constituents of the thesaurus
  - unigrams: broad, survey
  - bigrams: broad+survey
- We represent each lexical element **u** using a feature vector where:
  - Other lexical elements that co-occur with **u** in any review sentence are selected as features
    - *survey* and *broad+survey* are features for *broad*
  - unigrams and bigrams of the part of speech tags are also considered as features
    - *JJ* and *JJ+CC* are features for *broad*

Creating a Sentiment Sensitive Thesaurus (2/3)

Excellent and broad survey of the development of civilization
Excellent/JJ and/CC broad/JJ survey/NN1 of/IO the/AT development/NN1 of/IO civilization/NN1

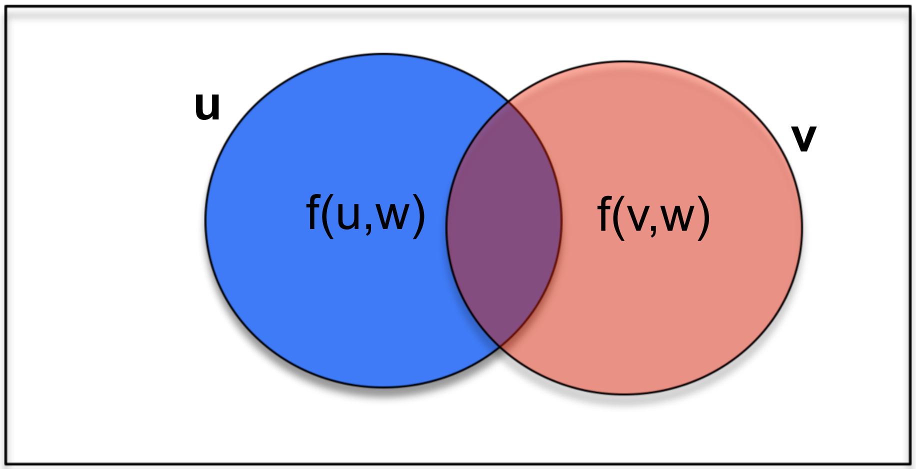
- We assign the document level sentiment labels (P: positive, N: negative) to each of the lexical elements generated from that document to create sentiment features
- sentiment features (lemma)
  - *broad*P*, *broad+survey*P*
- sentiment features (POS)
  - *JJ*P*, *JJ+NN1*P*
- Sentiment features can be created only from source domain labeled reviews
- We aggregate features from all sentences where a particular lexical element occur to create the feature vector for that lexical element

Creating a Sentiment Sensitive Thesaurus (3/3)

- The weight, f(**u**,*w*), assigned to a feature *w* in the feature vector representing a lexical element *u* is set to the *pointwise mutual information* between *u* and *w*
- The relatedness, τ(*v*,*u*), between two lexical elements *u* and *v* is computed as the fraction of weights for w such that if f(**u**,*w*) > 0 then f(**v**,*w*) > 0

$$\tau(v, u) = \frac{\sum_{w \in \{x \mid f(\mathbf{v}, x) > 0\}} f(\mathbf{u}, w)}{\sum_{w \in \{x \mid f(\mathbf{u}, x) > 0\}} f(\mathbf{u}, w)}$$

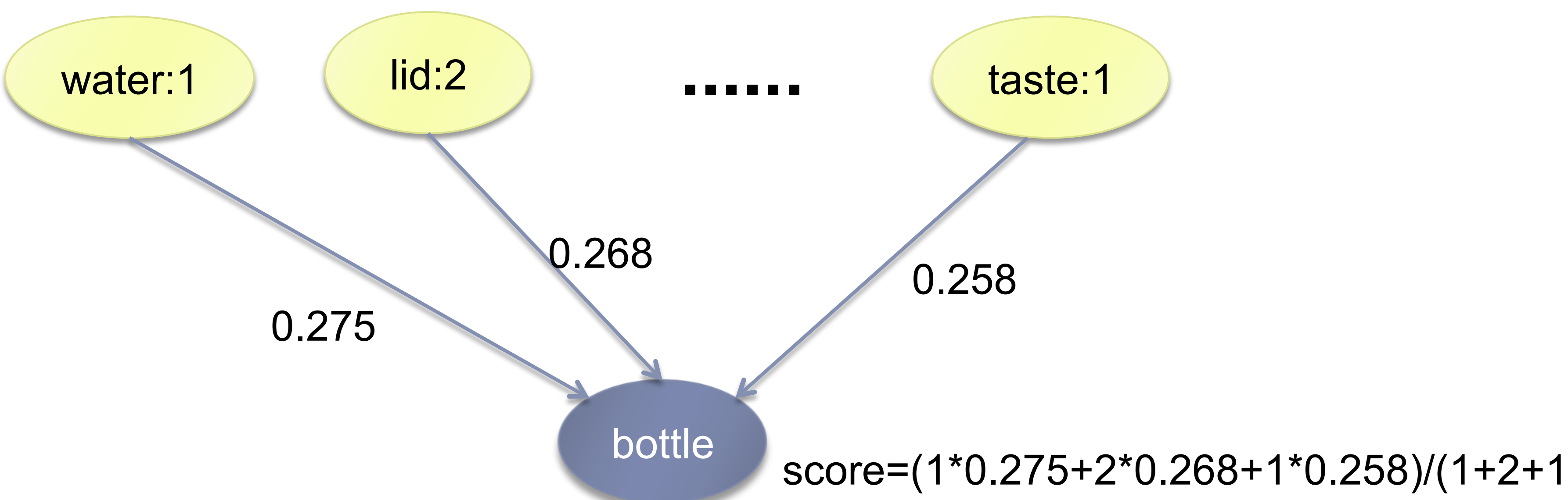
- Produce a sentiment sensitive thesaurus in which for each lexical element *u*, lexical elements *v* are listed in order of decreasing τ(*v*,*u*)



Representing a user review

- We represent a review *d* using a term-frequency vector **d**, where d_j is the number of occurrences of a lexical element w_j in *d*.
- We assign to each base entry u_i in the thesaurus a document-specific *score*(u_i, **d**) computed as the normalized sum of the product of $\tau(w_j, u_i)$ and the corresponding d_j frequency for each w_j in the document.

$$\text{score}(u_i, \mathbf{d}) = \frac{\sum_{j=1}^{N} d_j \tau(w_j, u_i)}{\sum_{l=1}^{N} d_l}$$



Feature Vector Expansion

- We expand **d** to **d'** by adding the top-scoring base entries from the thesaurus; these will be the base entries with many close neighbors present in the document.

<u>Sentiment Sensitive Thesaurus</u>
professional: excellent

(Because both words occur in similar contexts, e.g. *high+quality*P*)

We weight extended features by their rank to instead of the absolute value of the score because the differences between similarity scores is small in practice.
Extended features are assigned different feature ids to discriminate them from the *original* features.

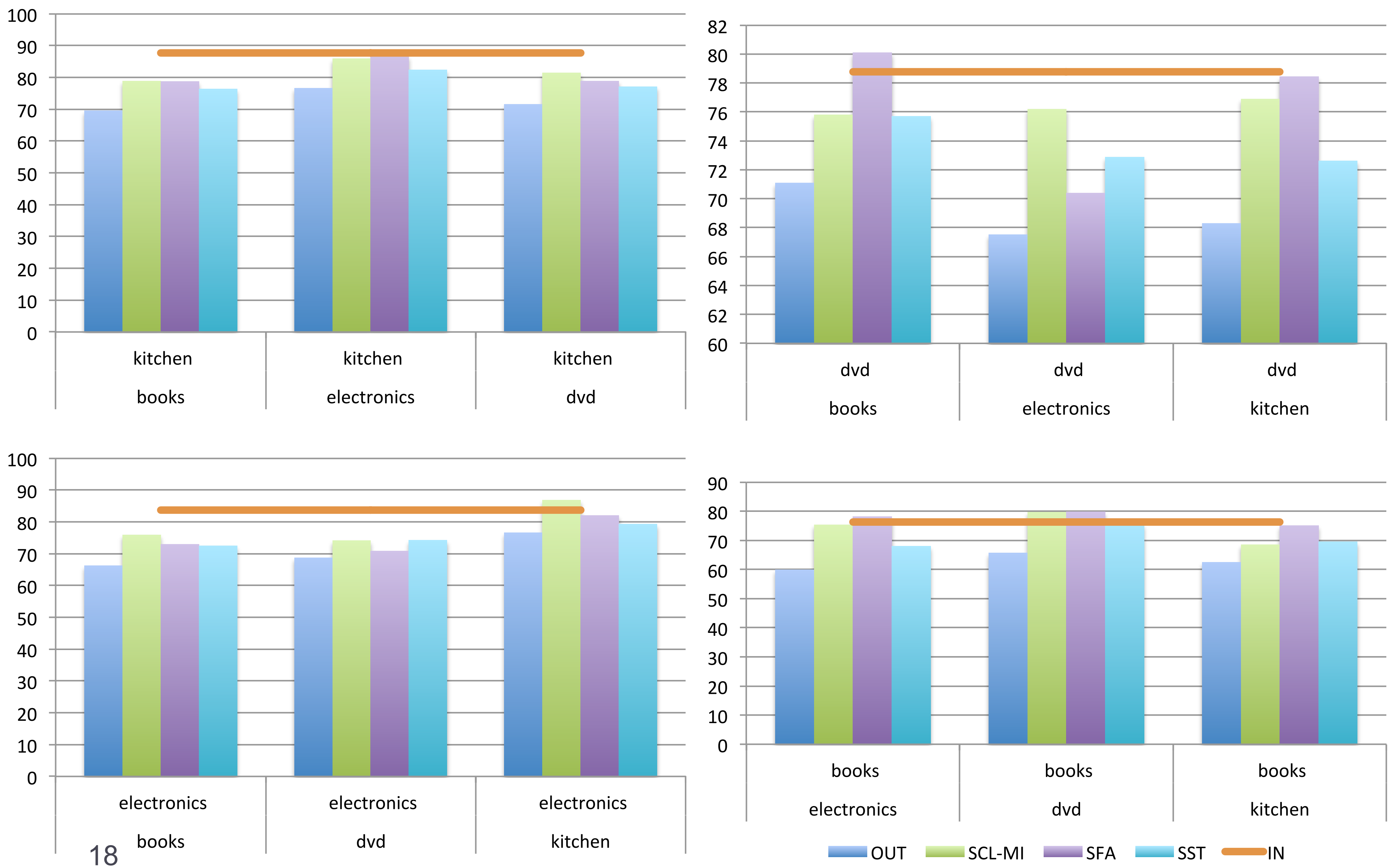
{..., (excellent, count(excellent,d)),..} +{...., (r-th ranked base entry, 1/rank),...}

original feature vector (**d**) + extended features = extended feature vector (**d'**)

Experimental Settings

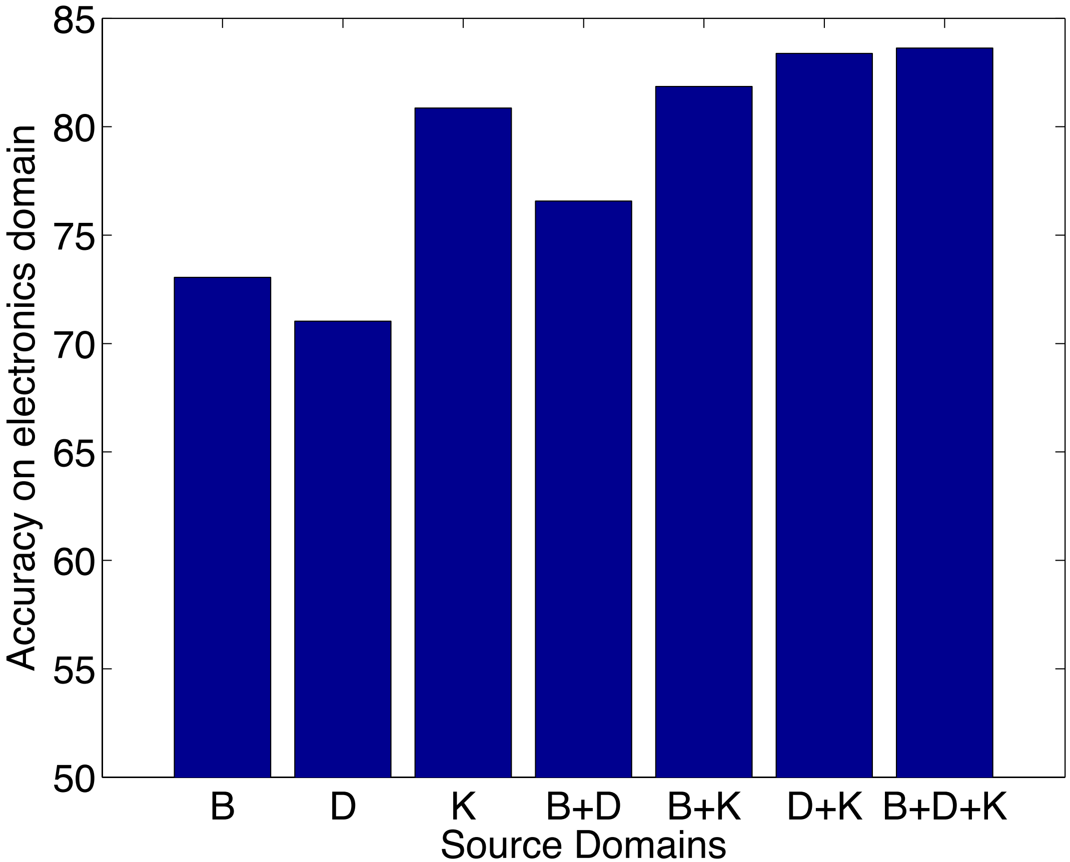
- Train a binary classifier on vectors **d'** using source domain labeled reviews
  - L2 regularized binary logistic regression is used as the classifier.
- Feature expansion is done at both train and test times
  - top scoring 1000 base entries are selected from the thesaurus
- We use the cross-domain sentiment classification dataset (Blitzer et al. 2007)
  - Four domains: Books (B), DVDs (D), electronic items (E), kitchen appliances (K)
  - 1000 positive, 1000 negative and on average 17,546 unlabeled reviews for each domain.
  - Randomly select 800 positive and 800 negative labeled reviews for each domain for training
  - Evaluate on the remaining 400 (200 positive and 200 negative).

Pairwise Domain Adaptation



More than one source domains...

- Keeping the no. of source domain training labeled instances constant, evaluate the performance when we combine multiple source domains.
- Sources = Books + DVDs + Kitchen
- Target = Electronics



Target domain
Electronics

Source domains
B: books
D: DVDs
K: kitchen appliances

Sentiment Classification Accuracy

Method	Kitchen	DVDs	Electronics	Books
No Adaptation *(out-of domain)*	72.61	68.97	70.53	62.72
NSS	77.50	73.50	75.50	71.46
Proposed	85.18	78.77	83.63	76.32
Target labeled (*in domain*)	*87.70*	*82.40*	*84.40*	*80.40*

No Thesaurus: lower bound

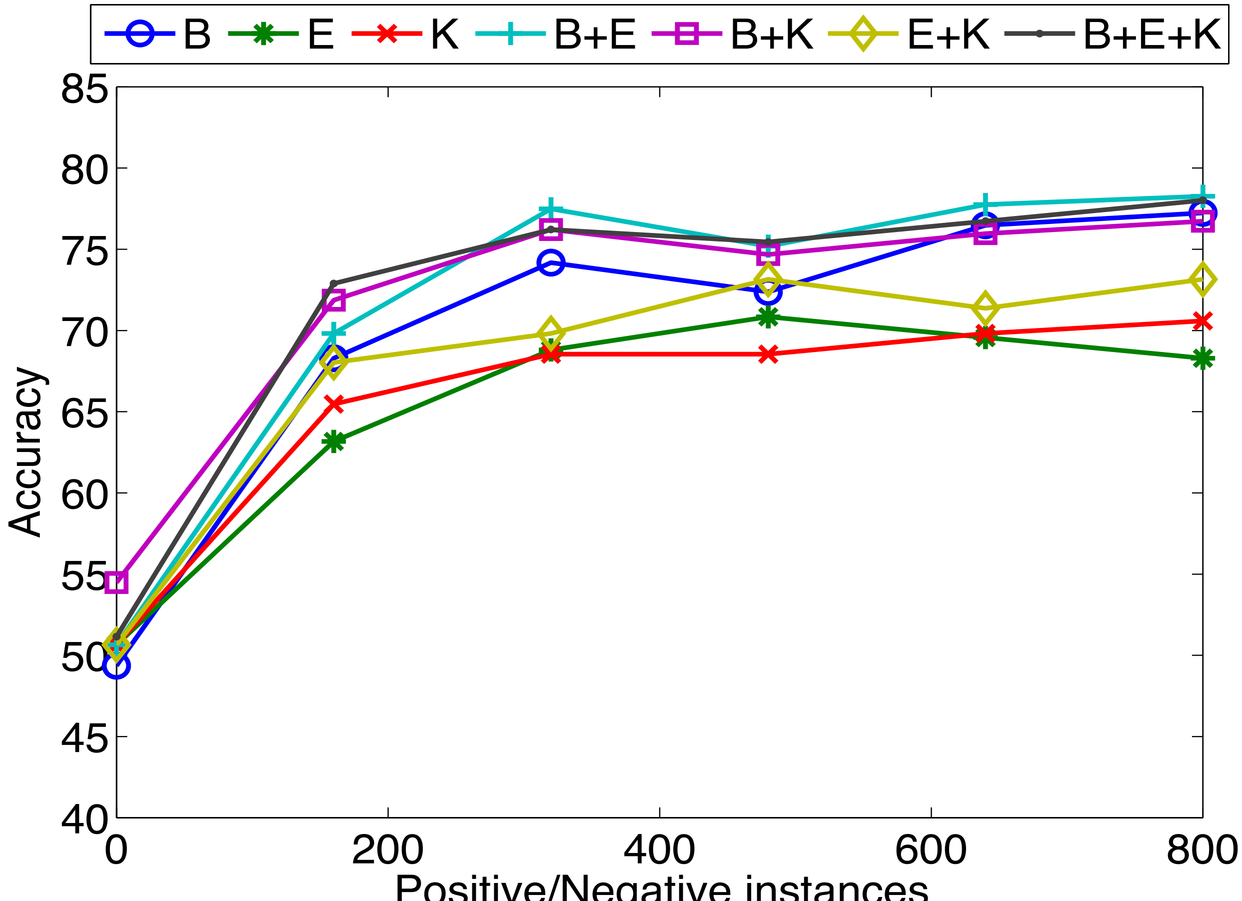
NSS: Non-sentiment sensitive thesaurus
(i.e. a thesaurus crated using the proposed method without any sentiment features)

Proposed: full sentiment thesaurus method

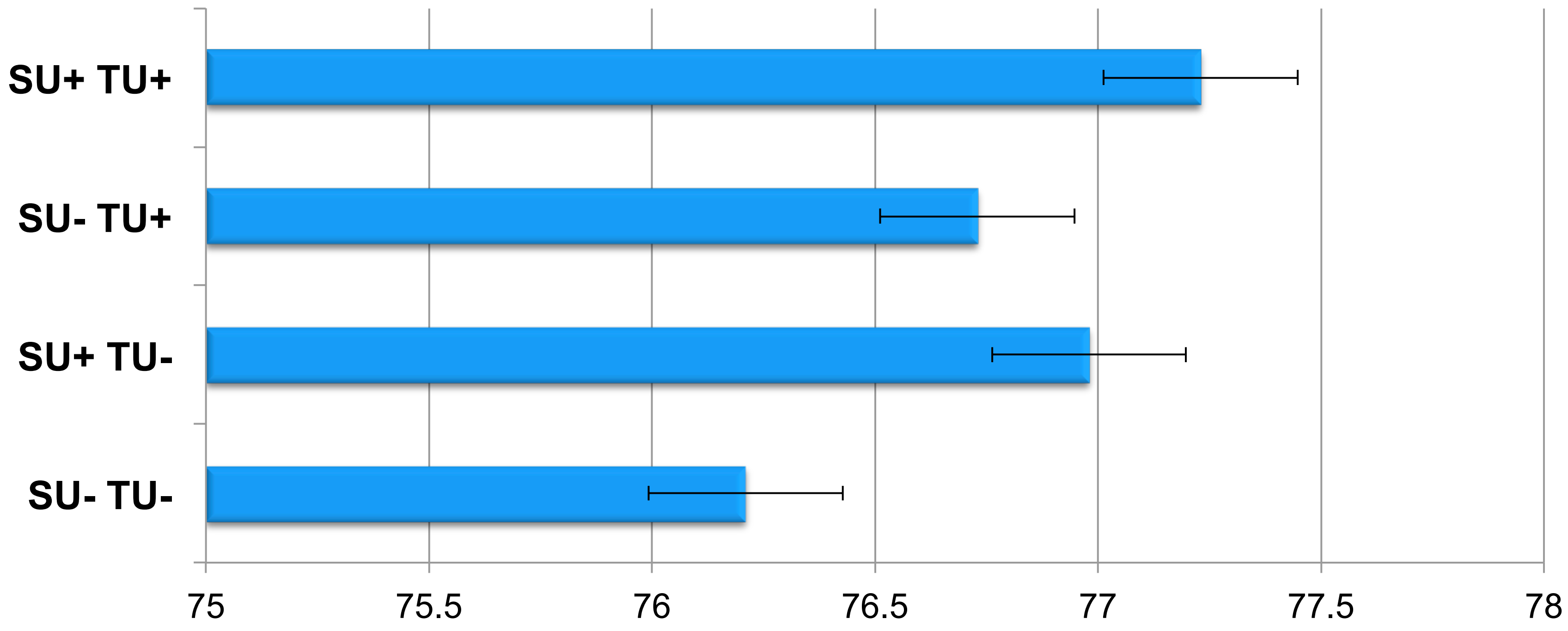
in domain: training on labeled target domain reviews

Effect of Source Domain Labeled Data

Gradually increase the amount of source domain labeled data (selected equal amounts when using multiple source domains) and measure the accuracy on the target (DVD) domain.



Effect of Source and Target Unlabeled Data



Source domains = Books + Electronics + Kitchen Appliances
Target domain = DVDs
SU (Source Unlabeled), TU (Target Unlabeled)

Conclusions and Current Work

- Sentiment sensitive thesaurus is useful to bridge the gap between source and target domain feature spaces
- By expanding feature vectors representing reviews at both test and train time we can overcome the feature mismatch problem in domain adaptation
- Combining multiple source domains improves sentiment classification accuracy on the target domain
- Amount of source domain labeled data has a direct impact on the performance
- Currently,
  - we are studying the performance of other relatedness measures and the use of target domain labeled data.
  - Use more than 3 source domains
- Data & Code
  - http://www.iba.t.u-tokyo.ac.jp/~danushka/data/SST.tgz

Thank You

Contact:
Email: danushka@iba.t.u-tokyo.ac.jp
Web: www.iba.t.u-tokyo.ac.jp/~danushka
Twitter: @Bollegala