Classifier Combination for Contextual Idiom Detection without Labelled Data

Linlin Li and Caroline Sporleder

MMCI / Computational Linguistics, Saarland University

{linlin,csporleder}@coli.uni-sb.de

EMNLP, Singapore
August 7, 2009

Why is Non-Literal Language a Problem?

Examples of Non-Literal Languagye

Dissanayake said that Kumaratunga was "playing with fire" after she accused military's top brass of interfering in the peace process. Kumaratunga has said in an interview she would not tolerate attempts by the army high command to sabotage her peace moves. A defence analyst close to the government said Kumaratunga had spoken a "load of rubbish" and the security forces would not take kindly to her disparaging comments about them.

Why is Non-Literal Language a Problem?

Examples of Non-Literal Languagye

Dissanayake said that Kumaratunga was "playing with fire" after she accused military's top brass of interfering in the peace process. Kumaratunga has said in an interview she would not tolerate attempts by the army high command to sabotage her peace moves. A defence analyst close to the government said Kumaratunga had spoken a "load of rubbish" and the security forces would not take kindly to her disparaging comments about them.

Non-Literal Expressions (idioms, metaphors etc.) ...

- occur frequently in language
- often behave idiosyncratically
- have to be recognised automatically to be analysed and interpreted in an appropriate way

Dealing with Idioms

Most previous research:

- automatic idiom extraction methods (type-based classification)

But:

- doesn't work for creative language use
- potentially idiomatic expressions can be used in literal sense

Literal Usage

(1) *Dad had to break the ice on the chicken troughs so that they could get water.*

(2) *Somehow I always end up spilling the beans all over the floor and looking foolish when the clerk comes to sweep them up.*

Dealing with Idioms

Most previous research:

- automatic idiom extraction methods (type-based classification)

But:

- doesn't work for creative language use
- potentially idiomatic expressions can be used in literal sense

Literal Usage

(1) *Dad had to break the ice on the chicken troughs so that they could get water.*

(2) *Somehow I always end up spilling the beans all over the floor and looking foolish when the clerk comes to sweep them up.*

⇒ **Idioms have to be recognised in discourse context!**
(token-based classification)

Token-based Idiom Classification

Previous Approaches:

- Katz and Giesbrecht (2006): supervised machine learning (k-nn), vector space model
- Birke and Sarkar (2006): bootstrapping from seed lists
- Cook et al. (2007), Fazly et al. (to appear): unsupervised, predict non-literal if idiom is in canonical form ($\approx$ dictionary form)
  - An idiomatic VNC (verb+noun combination) tends to have one (or at most a small number of) canonical form(s), which are its most preferred syntactic patterns (Fazly and Stevenson (2006))
  - This method determines the canonical form of an expression to be those forms whose frequency is much higher than the average frequency of all its forms

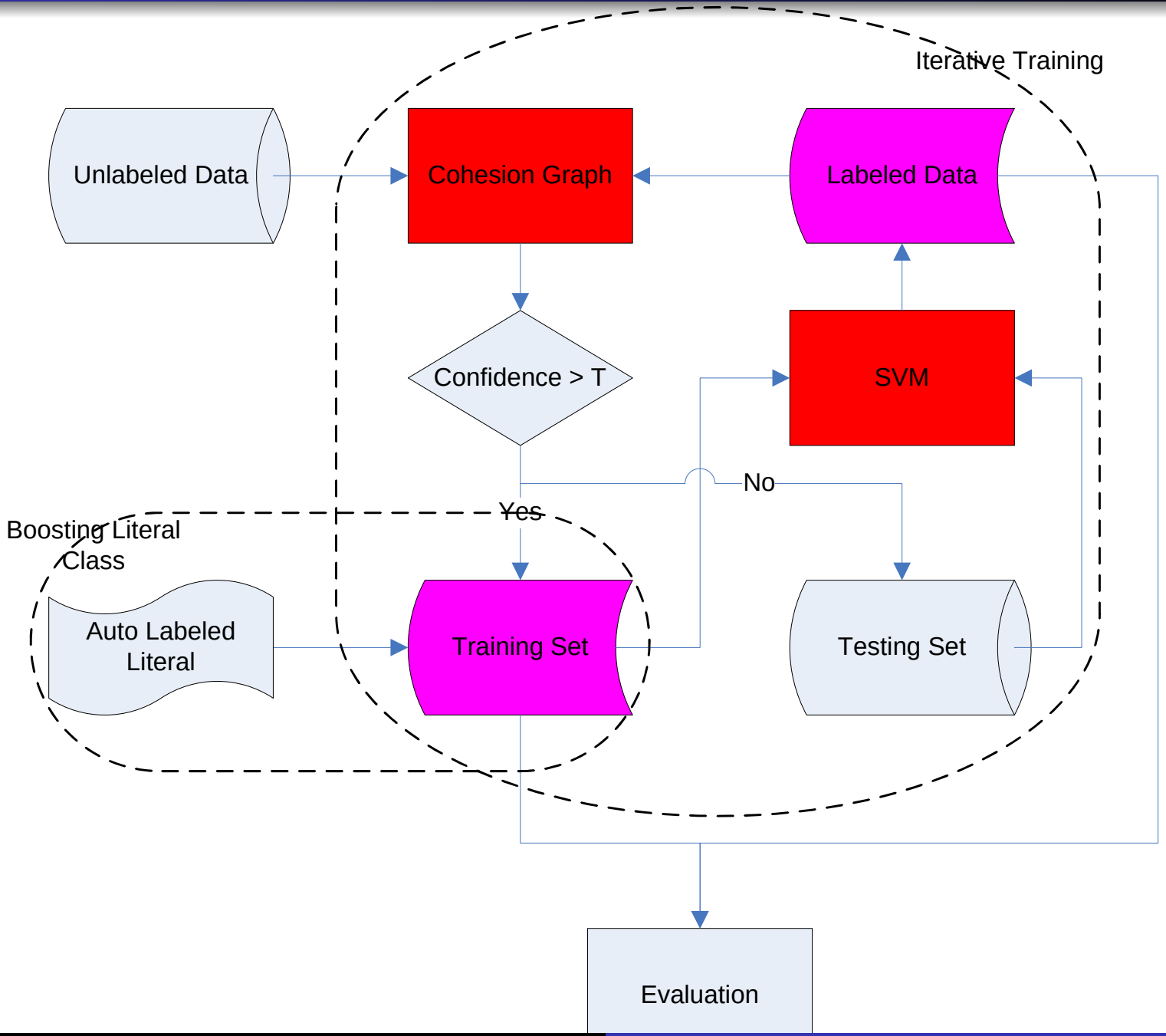
Token-based Idiom Classification

Previous Approaches:

- Sporleder and Li (2009): unsupervised method (cohesion graph) that exploits the presence or absence of cohesive ties between the component words of a potential idiom and its context
  - Component words of literally used expressions tend to exhibit lexical cohesion with their context, the words of non-literally used expressions do not
  - Based on the discourse connectivity change of the cohesion graph, which is defined as the average semantic relatedness change when excluding MWE component words
  - Modeling semantic relatedness: Normalised Google Distance (NGD) (Cilibrasi and Vitanyi, 2007)

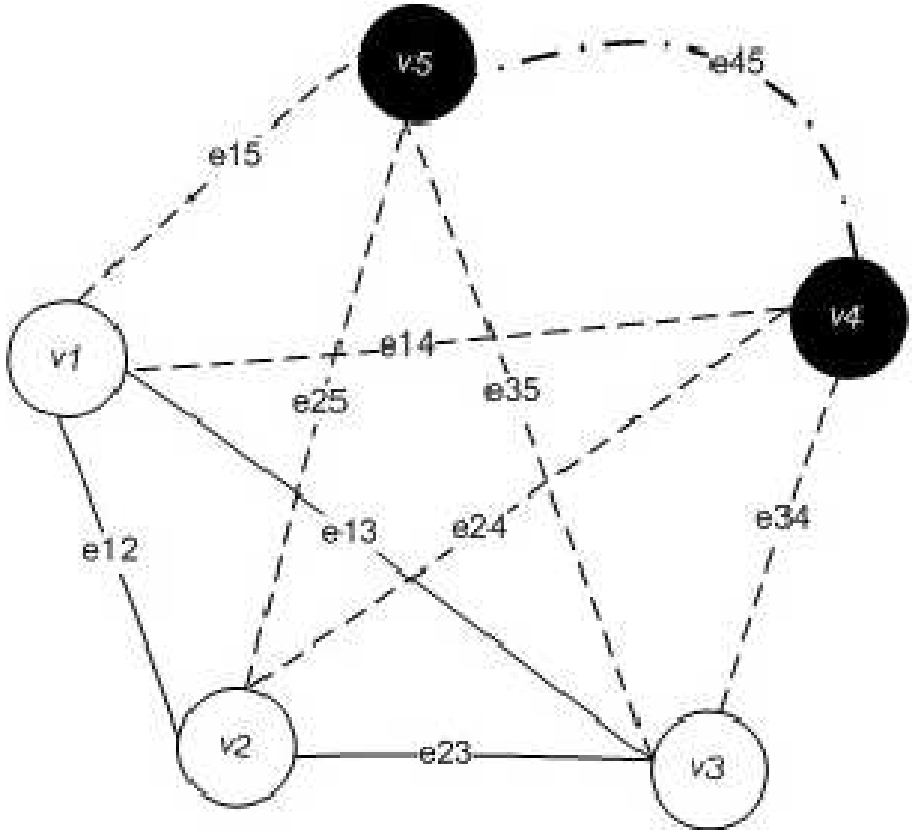
$\Rightarrow$ improvement on Sporleder and Li's (2009) results by employing a two-level strategy, in which a cohesion graph based unsupervised classifier is combined with a supervised classifier

Overview: Classifier Combination



Unsupervised Classifier: Cohesion Graph

We played$_{v1}$ a couple of party$_{v2}$ games$_{v3}$ to break$_{v4}$ the ice$_{v5}$.



- Graph-based Classifier ($\Delta c > 0 \Rightarrow$ literal):

$$\Delta c = c(G) - c(G')$$

$$(G : \{v1, v2, v3, v4, v5\},\ G' : \{v1, v2, v3\})$$

Problems with the Unsupervised Classifier

Many other linguistic clues are missing

- indicative prepositions (e.g.: between, over following break the ice)

Idiomatic Usages

(1) "Gujral will meet Sharif on Monday and discuss bilateral relations," the Press Trust of India added. The minister said Sharif and Gujral would be able to break the ice over Kashmir.

Problems with the Unsupervised Classifier

Many other linguistic clues are missing

- indicative prepositions (e.g.: between, over following break the ice)

Idiomatic Usages

(1) "Gujral will meet Sharif on Monday and discuss bilateral relations," the Press Trust of India added. The minister said Sharif and Gujral would be able to break the ice over Kashmir.

(2) Next week the two diplomats will meet in an attempt to break the ice between the two nations. A crucial issue in the talks will be the long-running water dispute.

Problems with the Unsupervised Classifier

Many other linguistic clues are missing

- indicative prepositions (e.g.: between, over following break the ice)
- idiomatic usages also exhibit cohesion with their context (e.g.: break the ice co-occurs with discuss, relations, talks, diplomacy)

Idiomatic Usages

(1) "Gujral will meet Sharif on Monday and discuss bilateral relations," the Press Trust of India added. The minister said Sharif and Gujral would be able to break the ice over Kashmir.

(2) Next week the two diplomats will meet in an attempt to break the ice between the two nations. A crucial issue in the talks will be the long-running water dispute.

Features (LIBSVM)

- **Salient words**: identify words which are particularly *salient* for literal usage, encode the frequencies of those words in the feature vectors

$$sal_{lit}(w) = \frac{\log f_{lit}(w) \times i_{lit}(w)}{\log f_{nonlit}(w) \times i_{nonlit}(w)}$$

  ($sal_{lit}(w)$: saliency score for the class *lit*; $f_{lit}(w)$: word frequency; $i_{lit}(w)$: instance frequency)
- **Related words**: encode the frequency of the top ranked words whose semantic relatedness with the noun in the idiomatic expression are highest
- **Relatedness score**: encodes the scores for the most highly connected edges in the cohesion graph
- **Discourse connectivity**: cohesion graph connectivity (i) when the target expression is included, $\underline{c(G)}$ (ii) when the target expression is excluded, $\underline{c(G')}$

Combining the Classifiers (1)

- Use the unsupervised classifier to label an initial training set for the supervised one
- Iteratively enlarging the training set
  - Only consider instances on whose labels both classifiers agree (reduce the noise)
  - Connectivity change of the unsupervised classifier is used as the confidence function
  - Re-training process involves re-computing the ranked lists of *salient and related words* and encode them in the feature vector

Problem: the iterative process introduce more and more imbalance in the training set

Combing the Classifiers (2)

Boosting the Literal Class

- Extract non-canonical form variants and label them as *literal* automatically
  - Change the number of the noun (*rock the boat* $\Rightarrow$ *rock the boats*)
  - Change the determiner (*rock a boat*)
  - Replace the verb or noun by one of its synonyms, hypernyms or siblings from WordNet (*rock the ship*)
- Add additional literal examples during each iteration

Experiments

Data

- 17 idioms (mainly V+NP and V+PP) with literal and non-literal sense
- all (canonical form) occurrences extracted from a Gigaword corpus (3964 instances)
- five paragraphs context
- manually labelled as "literal" (862 instances) or "non-literal" (3102 instances)

Experiments

Data (* = literal use is more common)

expression	literal	non-literal	all
back the wrong horse	0	25	25
bite off more than one can chew	2	142	144
bite one's tongue	16	150	166
blow one's own trumpet	0	9	9
bounce off the wall*	39	7	46
break the ice	20	521	541
drop the ball*	688	215	903
get one's feet wet	17	140	157
pass the buck	7	255	262
play with fire	34	532	566
pull the trigger*	11	4	15
rock the boat	8	470	478
set in stone	9	272	281
spill the beans	3	172	175
sweep under the carpet	0	9	9
swim against the tide	1	125	126
tear one's hair out	7	54	61
all	862	3102	3964

Results for Feature Analysis

Feature	Avg. literal (%) Prec.	Rec.	F-Score	Avg. (%) Acc.
salW	77.10	56.10	65.00	86.83
relW	78.00	43.20	55.60	84.99
relS	74.90	37.50	50.00	83.68
connectivity	78.30	2.10	4.10	78.58
salW+relW+relS	82.90	63.50	71.90	89.20
all	85.80	66.60	75.00	90.34

Results for Feature Analysis

Feature	Avg. literal (%) Prec.	Rec.	F-Score	Avg. (%) Acc.
salW	77.10	56.10	65.00	86.83
relW	78.00	43.20	55.60	84.99
relS	74.90	37.50	50.00	83.68
connectivity	78.30	2.10	4.10	78.58
salW+relW+relS	82.90	63.50	71.90	89.20
all	85.80	66.60	75.00	90.34

- The *salient words* feature has the highest performance

Results for Feature Analysis

Feature	Avg. literal (%) Prec.	Rec.	F-Score	Avg. (%) Acc.
salW	77.10	56.10	65.00	86.83
relW	78.00	43.20	55.60	84.99
relS	74.90	37.50	50.00	83.68
connectivity	78.30	2.10	4.10	78.58
salW+relW+relS	82.90	63.50	71.90	89.20
all	85.80	66.60	75.00	90.34

- The *salient words* feature has the highest performance
- *Connectivity gain* feature increases the performance of the model combined with the other features

Results for the Combined Classifier

Model	Acc.	Prec_l	Rec_l	F-Score_l
Base_{maj}	78.25	-	-	-
unsup.	78.38	50.04	69.72	58.26
combined	86.30	83.86	45.82	59.26
combined+boost	86.13	70.26	62.76	66.30
combined+it*	86.68	85.68	46.52	60.30
combined+boost+it*	**87.03**	**71.86**	**66.36**	**69.00**
super. 10CV	90.34	85.80	66.60	75.00

- **Base_{maj}**: majority baseline, i.e., "non-literal" (cf. CForm classifier by Cook et al. (2007), Fazly et al. (to appear))
- **unsup.**: cohesion graph (Sporleder and Li (2009))
- **combined**: combined classifier
- **combined + boost**: combined classifier with boosting literal class
- **combined + +boost + it***: iteratively increase trainning set, boosting literal class in each iteration
- **super.10CV**: 10-fold cross validation for supervised classifier

Results for the Combined Classifier

Model	Acc.	Prec_l	Rec_l	F-Score_l
Base_{maj}	78.25	-	-	-
unsup.	78.38	50.04	69.72	58.26
combined	86.30	83.86	45.82	59.26
combined+boost	86.13	70.26	62.76	66.30
combined+it*	86.68	85.68	46.52	60.30
combined+boost+it*	**87.03**	**71.86**	**66.36**	**69.00**
super. 10CV	90.34	85.80	66.60	75.00

- **Base_{maj}**: majority baseline, i.e., "non-literal" (cf. CForm classifier by Cook et al. (2007), Fazly et al. (to appear))
- **unsup.**: cohesion graph (Sporleder and Li (2009))
- **combined**: combined classifier
- **combined + boost**: combined classifier with boosting literal class
- **combined + +boost + it***: iteratively increase trainning set, boosting literal class in each iteration
- **super.10CV**: 10-fold cross validation for supervised classifier

Results for the Combined Classifier

Model	Acc.	Prec_l	Rec_l	F-Score_l
Base_{maj}	78.25	-	-	-
unsup.	78.38	50.04	69.72	58.26
combined	86.30	83.86	45.82	59.26
combined+boost	86.13	70.26	62.76	66.30
combined+it*	86.68	85.68	46.52	60.30
combined+boost+it*	**87.03**	**71.86**	**66.36**	**69.00**
super. 10CV	90.34	85.80	66.60	75.00

- **Base_{maj}**: majority baseline, i.e., "non-literal" (cf. CForm classifier by Cook et al. (2007), Fazly et al. (to appear))
- **unsup.**: cohesion graph (Sporleder and Li (2009))
- **combined**: combined classifier
- **combined + boost**: combined classifier with boosting literal class
- **combined + +boost + it***: iteratively increase trainning set, boosting literal class in each iteration
- **super.10CV**: 10-fold cross validation for supervised classifier

Results for the Combined Classifier

Model	Acc.	Prec_l	Rec_l	F-Score_l
Base_{maj}	78.25	-	-	-
unsup.	78.38	50.04	69.72	58.26
combined	86.30	83.86	45.82	59.26
combined+boost	86.13	70.26	62.76	66.30
combined+it*	86.68	85.68	46.52	60.30
combined+boost+it*	**87.03**	**71.86**	**66.36**	**69.00**
super. 10CV	90.34	85.80	66.60	75.00

- **Base_{maj}**: majority baseline, i.e., "non-literal" (cf. CForm classifier by Cook et al. (2007), Fazly et al. (to appear))
- **unsup.**: cohesion graph (Sporleder and Li (2009))
- **combined**: combined classifier
- **combined + boost**: combined classifier with boosting literal class
- **combined + +boost + it***: iteratively increase trainning set, boosting literal class in each iteration
- **super.10CV**: 10-fold cross validation for supervised classifier

Results for the Combined Classifier

Model	Acc.	Prec_l	Rec_l	F-Score_l
Base_{maj}	78.25	-	-	-
unsup.	78.38	50.04	69.72	58.26
combined	86.30	83.86	45.82	59.26
combined+boost	86.13	70.26	62.76	66.30
combined+it*	86.68	85.68	46.52	60.30
combined+boost+it*	**87.03**	**71.86**	**66.36**	**69.00**
super. 10CV	90.34	85.80	66.60	75.00

- **Base_{maj}**: majority baseline, i.e., "non-literal" (cf. CForm classifier by Cook et al. (2007), Fazly et al. (to appear))
- **unsup.**: cohesion graph (Sporleder and Li (2009))
- **combined**: combined classifier
- **combined + boost**: combined classifier with boosting literal class
- **combined + +boost + it***: iteratively increase trainning set, boosting literal class in each iteration
- **super.10CV**: 10-fold cross validation for supervised classifier

Results for the Combined Classifier

Model	Acc.	$Prec_l$	Rec_l	$F\text{-}Score_l$
$Base_{maj}$	78.25	-	-	-
unsup.	78.38	50.04	69.72	58.26
combined	86.30	83.86	45.82	59.26
combined+boost	86.13	70.26	62.76	66.30
combined+it*	86.68	85.68	46.52	60.30
combined+boost+it*	**87.03**	**71.86**	**66.36**	**69.00**
super. 10CV	90.34	85.80	66.60	75.00

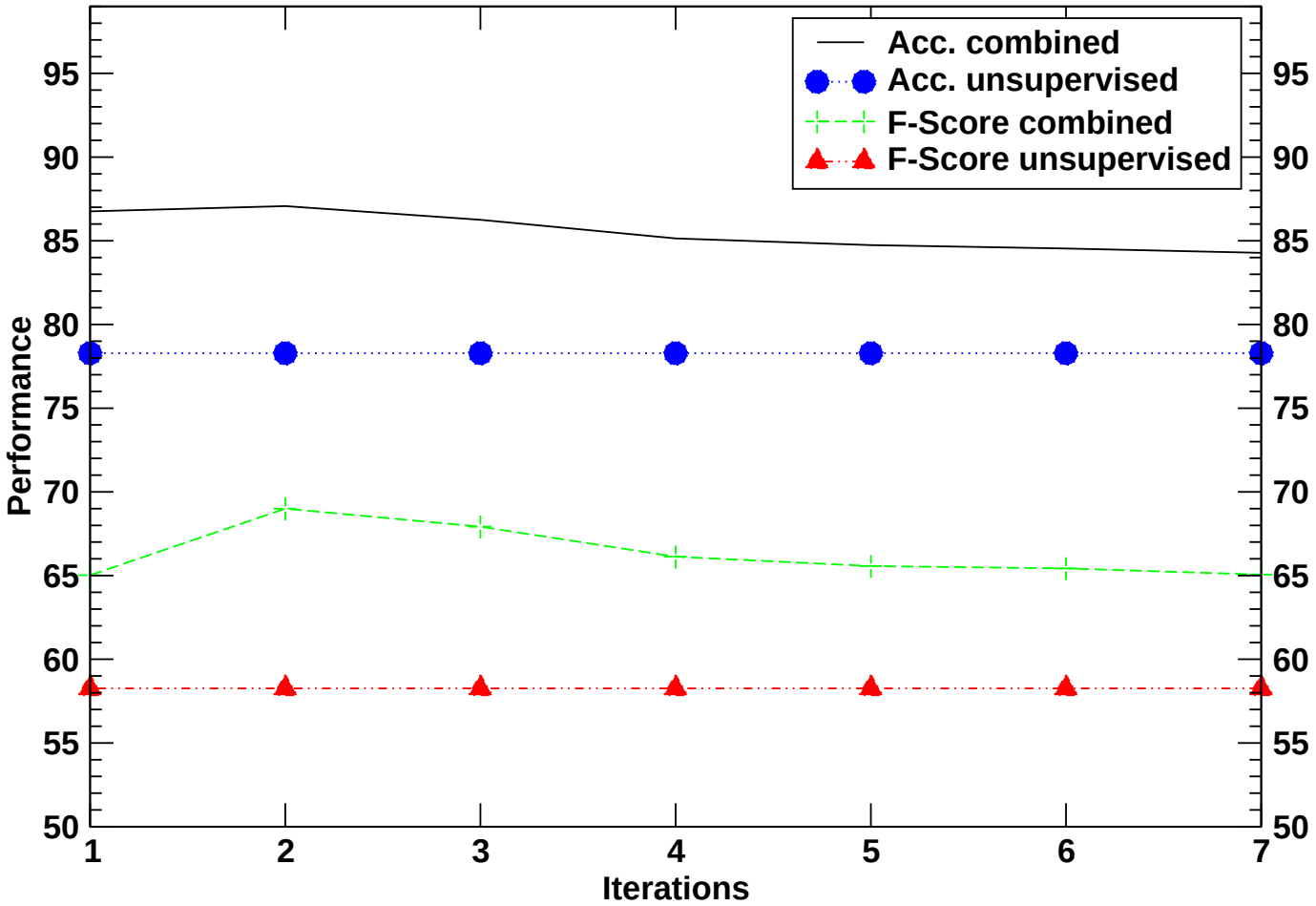
- **$Base_{maj}$**: majority baseline, i.e., "non-literal" (cf. CForm classifier by Cook et al. (2007), Fazly et al. (to appear))
- **unsup.**: cohesion graph (Sporleder and Li (2009))
- **combined**: combined classifier
- **combined + boost**: combined classifier with boosting literal class
- **combined + +boost + it***: iteratively increase trainning set, boosting literal class in each iteration
- **super.10CV**: 10-fold cross validation for supervised classifier

Results for the Combined Classifier

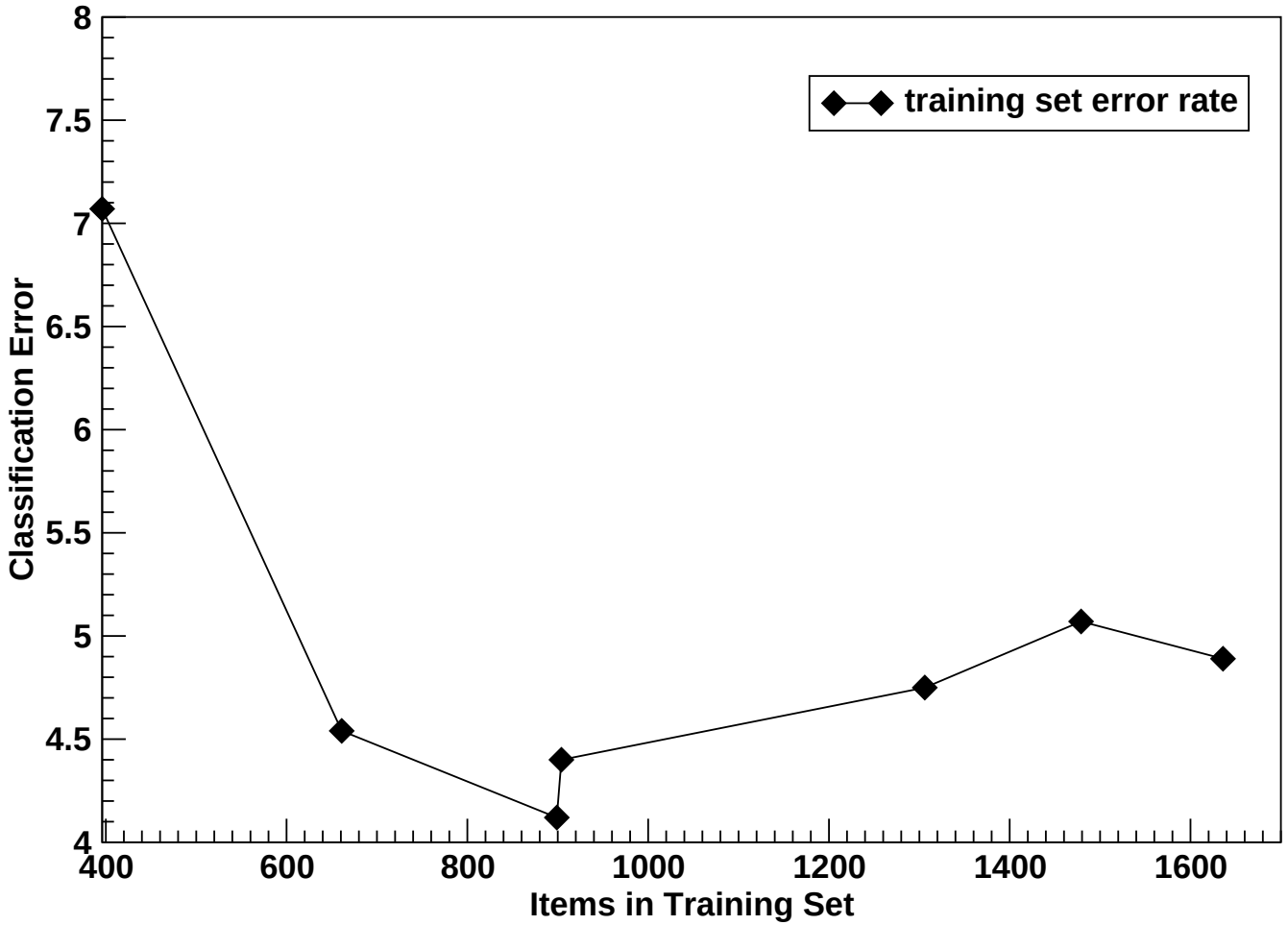
Model	Acc.	$Prec_l$	Rec_l	$F\text{-}Score_l$
$Base_{maj}$	78.25	-	-	-
unsup.	78.38	50.04	69.72	58.26
combined	86.30	83.86	45.82	59.26
combined+boost	86.13	70.26	62.76	66.30
combined+it*	86.68	85.68	46.52	60.30
combined+boost+it*	**87.03**	**71.86**	**66.36**	**69.00**
super. 10CV	90.34	85.80	66.60	75.00

- **$Base_{maj}$**: majority baseline, i.e., "non-literal" (cf. CForm classifier by Cook et al. (2007), Fazly et al. (to appear))
- **unsup.**: cohesion graph (Sporleder and Li (2009))
- **combined**: combined classifier
- **combined + boost**: combined classifier with boosting literal class
- **combined + +boost + it***: iteratively increase trainning set, boosting literal class in each iteration
- **super.10CV**: 10-fold cross validation for supervised classifier

Iterative Training with Boosting Literal Class



Error in the Training Set



Conclusion and Future Work

- Completely unsupervised method, which complements an unsupervised classifier with a supervised classifier, which explores lexical cohesion features and other linguistic clues
- The combined classifier can lead to significant reduction of classification errors
- Performance can be improved further by boosting the literal cases, which can be automatically extracted from an unlabeled corpus
- In the future, experiment with linguistically more informed features to improve the supervised classifier