Identifying Event Descriptions using Co-training with Online News Summaries

William Yang Wang and **Kapil Thadani** and **Kathleen R. McKeown**
Computer Science Department
Columbia University
yww@cs.cmu.edu, {kapil, kathy}@cs.columbia.edu

Abstract

Systems that distill information about events from large corpora generally extract sentences that are relevant to a short event query. We present a novel co-training strategy for this task that employs a multi-document news summary corpus featuring 2.5 million unlabeled sentences, thus obviating the need for extensive manual annotation. Our experiments indicate that this technique significantly outperforms standard classification approaches with linear feature combination on this task. An analysis of our approach under various settings reveals how classifier and parameter choice can be used to control runtime overhead while contributing to an absolute increase of 22% in recall.

1 Introduction

Automatic identification of event descriptions is a crucial, yet difficult, problem with impact on applications such as question answering (QA), query-focused summarization (QS), and text mining (TM) systems. In such applications, a system is given a description of an event in the form of a query and the task is to identify sentences within relevant documents (retrieved by an information retrieval system) that describe the event. We define *event-relevant* sentences as those describing a unique event as specified in the query, typically occurring at a specific location, on a specific date, or with specific participants. The task is made difficult by the fact that words in the query rarely provide enough identifying information (Xu and Croft, 2000; Chirita et al., 2007) to reliably find descriptive sentences; the query may not specify all named entities and may provide other descriptive terms that are not easily matched. For example, given the query requesting a description of *"the Israeli-Palestinian peace talks"*, the sentence *"The discussions between Israelis and Palestinians follow last month's Annapolis meeting where Israeli Prime Minister and Palestinian President Mahmoud Abbas met and agreed to try to negotiate a deal before the end of 2008."* is event-relevant, whereas the sentence *"Turkey has close ties to both Israel and the Palestinians."* is not. Yet both sentences feature the named entities from the query. The term "peace talks" does not appear in the event-relevant sentence, but is implied by "negotiate".

Previous approaches have addressed this problem using supervised learning (Mani et al., 2003; Bethard and Martin, 2006; Manshadi et al., 2008), where the relation between the query and event-relevant sentences is learned. Still, gathering a large amount of training data is difficult and people often do not agree on what counts as relevant to an event description (Filatova and Hatzivassiloglou, 2003), making the process of manually labeling sentences as events both time consuming and expensive. This is supported by our own experiments with Amazon Mechanical Turk (AMT); we were only able to obtain 685 labeled event-relevant sentences on which 3 or more AMT users agreed after 16 days of posting. This is a striking contrast with use of AMT for other labeling tasks where thousands of labeled examples can be obtained in a matter of hours (Snow et al., 2008; Marge et al., 2010; Rosenthal et al., 2010).

In this paper, we present a novel semi-supervised learning method using co-training to classify sentences as salient event descriptions for a given event query. We use a small amount of manually annotated seed data, in the form of (query, sentence) pairs. Critically, we then augment this with a large amount of unlabeled news summaries, spanning nine years, from the Web. We hypothesize that the headline of each summary can be viewed as a single event query, and the corresponding summary sentences include a good number of salient event sentences that directly describe the event, as well as a small number of non-event sentences. We note that such data is abundantly available on the Web given naturally-occurring news stories as well as mature multi-document summarization systems. We use two relatively simple sets of features, one based on keywords and the other based on named entities, to train two Bayesian network classifiers on seed training data and employ co-training over the collected news summaries to incrementally augment the training set with increasingly robust labeled examples. Experimental results show a significant, absolute gain of 8% over training of the classifiers on the seed data alone.

Our primary contributions in this paper include:

- Collection of an online news summary corpus for detecting event relevance[1] that includes 166,435 summaries (2.5M sentences) generated by an online news summarizer in the period 2003-2011.
- An efficient co-training strategy for identifying event descriptions using online news summaries and compact, simple feature sets.
- An evaluation of the impact of classification techniques, experimental parameters and amount of novel Web data on the accuracy and efficiency of co-training.

In the following sections, we first present related work on event identification and then present our approach, outlining our hypothesis, the data we used, the co-training strategy and feature sets for this task. Following this, we present experimental results and conclude with a discussion of the implications and limitations of this work.

2 Related Work

The problem of identifying and understanding events in natural language text has been explored in many ways that have produced a variety of perspectives on the challenges involved. Tasks explored have included the mapping of verb-level events into aspectual classes (Siegel and McKeown, 2000), detection of specific *atomic* events (Filatova and Hatzivassiloglou, 2003), supervised event classification (Bethard and Martin, 2006) and unsupervised learning of event schemas (Chambers and Jurafsky, 2009). The notion of what constitutes an event has similarly ranged across perspectives, from general verb-level events (Siegel and McKeown, 2000; Chambers and Jurafsky, 2009) and predicate-argument pairs (Manshadi et al., 2008) to more specific news events (Spitters and Kraaij, 2002; Filatova and Hatzivassiloglou, 2003) and public health events (Fisichella et al., 2010).

Our task also adheres to a more specific definition of events: we assume that event queries refer to unique and newsworthy events such "the Beijing Olympics", rather than more general lexical-level events such as "the game". Our sentence-retrieval task aligns with the perspective of Filatova and Hatzivassiloglou (2003) who defined the notion of *atomic* events at the sentence level. They also show how event identification at this level can benefit indexing, summarization, and QA using low-level lexical features (Filatova and Hatzivassiloglou, 2004).

Many event identification tasks rely on supervised learning methods (Siegel and McKeown, 2000; Mani et al., 2003; Bethard and Martin, 2006; Manshadi et al., 2008) that can incur high annotation costs. While fully unsupervised methods (Bejan, 2008; Spitters and Kraaij, 2002) for event discovery are not encumbered by the availability of training data, they can produce event clusters or topics which are difficult to interpret. In contrast, our semi-supervised learning approach balances the pros and cons of both supervised and unsupervised approaches.

3 Our Approach

We frame the task of selecting event-relevant sentences as a binary classification problem over all sentences in the corpus. The following sections detail the data, features and techniques used.

3.1 Corpora

Seed Training Data and Test Data Annotation

Our primary corpus of news documents and queries is derived from the DARPA Global Autonomous Language Exploitation (GALE) distillation training and evaluation sets. Fifty event queries provided for the GALE task were used for our experiments (randomly divided into 30 training queries and 20 test queries), and the set of documents was restricted to those containing at least one keyword bigram from any query in the set[2]. For each query, we consider all sentences from the

[1] http://www.cs.columbia.edu/~kathy/Data/CSC.tar.gz

[2] As determined by the information-retrieval pipeline built using Lucene: `http://lucene.apache.org`

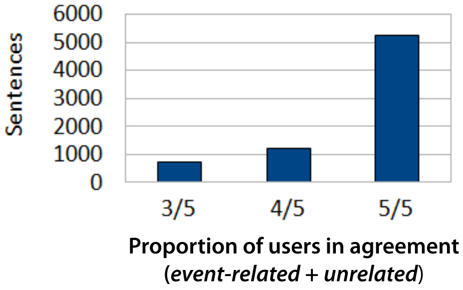

Figure 1: Number of sentences where a proportion of AMT users agreed on the classification (*event-related/unrelated*). With a binary task, 1/5 implies 4/5 agreement and 2/5 implies 3/5 agreement.

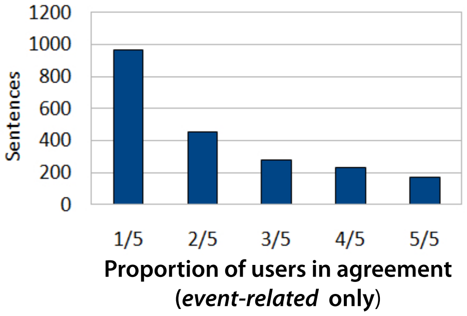

Figure 2: Number of sentences where a majority of AMT users agreed on the classification *event-related*.

top 10 retrieved documents and resort to AMT[3] to obtain sentence-level labels.

Each AMT annotation task presented AMT users with four contiguous sentences in conjunction with the query and asked them to indicate whether each of the individual sentences was relevant to the query[4]. The users could also view the entire document for context. Each task was assigned to 5 AMT users and a total of 252 unique users from the United States participated in the tasks. The full study lasted 16 days and had an overall cost of $350.

Fig. 1 shows the number of sentences (out of a total of 7161 candidates for all 50 queries) whose classification was agreed upon by a certain proportion of annotators. This result appears to suggest that annotators often agree when labeling event-relevant sentences[5]. However, these numbers are dominated by the majority class (irrelevant sentences), indicating that it is relatively easy for AMT users to agree when a sentence is not relevant. This is confirmed by Fig. 2, which displays the number of sentences that were specifically tagged as event-relevant for their respective queries. The breakdown for event-relevant sentences shows that about 30% of the sentences were identified as event-relevant by at least one AMT user and only a third of these received a majority vote for the label. This skewed distribution poses a significant challenge to the construction of a balanced training corpus for this task at low cost. Our observations are commensurate with Filatova and Hatzivassiloglou (2003) who note that event annotation is difficult for human annotators.

Unlabeled News Summary Data

Since manual creation of a large set of event-relevant sentences is difficult, we make use of a semi-supervised technique that employs a large quantity of unlabeled data to iteratively augment the small corpus described above. Unlabeled training data for our task also must comprise event queries and groups of sentences that are both related and unrelated to these events.

We experiment with automatically-generated news summaries as unlabeled training data for identifying event-related sentences. We postulate that the title of each news summary forms a single query and assume that at least some of the sentences in the summary will relate to the event mentioned in the title, while some may not. The choice of *summaries* of online news documents rather than the documents is prompted by the assumption that the distribution of query-related sentences and query-unrelated sentences is more balanced in short, informative summaries; this allows us to use the data more efficiently and avoid the sparse occurrence of event-specific sentences in online news articles.

The unlabeled dataset used in this work was retrieved from the output of an online news summarization system, Newsblaster http://newsblaster.cs.columbia.edu/, that crawls the Web for news articles, clusters them on specific topics and produces multidocument summaries for each cluster. We collected a total of 166,435 summaries containing 2.5 million sentences and covering 2,129 days in the 2003-2011 period.

As an initial experiment in augmenting the training corpus, we assumed that *all* summary sentences were relevant to the title query and trained

[3] `http://mturk.com`

[4] We also added an obviously fake sentence to detect sloppy annotators and robots.

[5] Fleiss' $\kappa = 0.417$, generally assumed to indicate moderate agreement between annotators

classifiers over a balanced corpus created by directly adding some summary data to the seed corpus. We observed the performance of these classifiers was poorer than that of classifiers trained only on the unaugmented seed corpus, suggesting that the assumption was too strong and that the summaries consist of both query-related and query-unrelated sentences. This supports the use of such a corpus in a semi-supervised setting.

3.2 Semi-Supervised Approach

The original co-training framework (Blum and Mitchell, 1998) was introduced in the context of Web page classification where one typically has access to a limited number of labeled pages but to a potentially unlimited number of unlabeled pages. Co-training involves building two independent views (classifiers), letting them automatically label the unlabeled data and incorporating this self-labeled data into the seed training set to improve system performance. Mihalcea (2004) shows that co-training is useful in word sense disamgibuation, Wan (2009) uses SVM for co-training and shows improvement for cross-lingual sentiment analysis, and recent research has established the effectiveness of co-training for a variety of NLP tasks (Yu and Kübler, 2011; Li et al., 2011; Bergsma et al., 2011). In this section, we present our co-training classifiers, as well as an algorithm that is specifically tailored to automatic event identification.

Features

Implementation of the co-training algorithm involves the design of two independent views (classifiers) for the same instance. In the original co-training algorithm for Web page classification, Blum and Mitchell (1998) use hyperlinks and bag-of-words as the approximation[6] of two distinct views of a Web page. In our approach, we choose two simple views to represent each candidate sentence for a given event query: (1) keyword features that include unigram and bigram overlaps between a candidate sentence and the event query. (2) named entity features including the number of exact entity (*Person*, *Location*, and *Organization*) matches between a candidate sentence and the query, as well as the total number of co-occurring named entity tags (regardless of the actual entity being tagged) between a candidate sentence and the query. We used the Stanford Named Entity tagger (Finkel et al., 2005) to obtain named entities features from the unlabeled data.

Note that the second view of our approach is clearly not independent of the first view, as the named entity matching is based on lexical matching. However, named entities are known as an informative source for selecting relevant sentences for a query since they serve as unique identifiers (Parton et al., 2008). Previous studies (Krogel and Scheffer, 2004) show that if the independence assumption of co-training approach is violated, the co-training approach can yield negative results. However, our results show that co-training yields improvement even though are classifiers are not independent.

The Co-training Algorithm

The motivation behind our co-training algorithm is to make use of the online summarization data for event identification, and improve recall as well as precision.The pseudo code of this algorithm is provided in Algorithm 1.

Algorithm 1 The co-training algorithm

```
Given:
(1) a set $S$ of labeled seed training examples;
(2) a set $U$ of unlabeled news summary examples;

Initialize the iteration parameter $k$, pool size $u$, and leap size $v$;
for $i = 1 \rightarrow k$ do
    Create a temporary pool by randomly choosing $u$ examples from $U$;
    Use $L$ to train a classifier $C_1$ using only keyword features;
    Use $L$ to train a classifier $C_2$ using only named entity features;
    Run $C_1$ to label $u/2$ examples and select $v/2$ balanced examples;
    Run $C_2$ to label $u/2$ examples and select $v/2$ balanced examples;
    Add all selected examples to $L$;
    Return the remaining examples back to $U$;
end for
```

Earlier work (Mihalcea, 2004) indicates that by choosing only high confidence examples from the self-labeled data set, we can improve the co-training results. However, we disagree with this assumption; if low-confidence examples are re-

[6] Note that it is very difficult to prove that two views are completely independent of each other. (Du et al., 2010)

moved from training, the final co-trained classifier will have problems evaluating difficult examples in the held-out data set [7].

Our algorithm is a variation of the original co-training algorithm (Blum and Mitchell, 1998). In the original algorithm, p represents the number of self-labeled positive cases and n represents the number of self-labeled negative cases in each iteration. We now eliminate the p and n parameters, and use v to measure the *leap size*, the amount of self-labeled data added to the training set in each iteration. This is to ensure that, in each iteration, balanced selection of unlabeled examples from the two views can be added to the labeled data. We keep the unselected examples and return them to the pool U at the end of each iteration to make sure we use the unlabeled data exhaustively.

Bayesian Network Classifier

We employ a Bayesian network classification scheme for the co-training experiments described in section 4. Previous approaches to co-training have successfully used the well-known naive Bayes classification scheme (Blum and Mitchell, 1998; Mihalcea, 2004), which assumes conditional independence between features.

A Bayesian network approach is a variation on this probabilistic classification scheme that avoids making strong independence assumptions between features. Building the classifier entails an additional step for estimating the conditional dependencies between the features; this is accomplished by using search algorithms and scoring metrics[8] to learn a DAG structure representing a Bayesian network over the features. Once such structure is estimated, conditional probabilities $p(x_i|\pi_i)$ are estimated[9] where π_i represents the set of features that feature x_i is assumed to conditionally depend on. Classification of an unseen example $\mathbf{x}'$ is performed similarly to the naive Bayes approach with the estimation of $\arg\max_y p(y|\mathbf{x}')$.

4 Experiments

We present four experiments to test system performance. First, we present the comparisons between our approach and other supervised/semi-supervised baselines, arbitrarily choosing settings for the number of co-training iterations k and the number of sentences added in each iteration[10]. We then test the trade-off between k and v by fixing the total amount of unlabeled data available, i.e., $k * v$ sentences[11]. In the third experiment, we test the impact of leap size v on both accuracy and efficiency by fixing k, as well as the impact of the iteration parameter k by fixing v. In the final experiment, we empirically choose the best possible v to test the influence of unlabeled data size on the recall and precision of our system. Due to the randomness in our co-training algorithm, we repeat every experiment 5 times and report the average results.

4.1 Comparing with Baseline Approaches

Table 1 shows the overall Bayesian network co-training results in comparison to various baseline systems. Previous work (Blum and Mitchell, 1998; Mihalcea, 2004; Wan, 2009) points out that naive Bayes and SVM classifiers can achieve promising results for co-training, so we use the same co-training settings ($k = 50$, $v = 500$) to compare these two classifiers with the Bayesian network classifier. Classifier 1 corresponds to the keyword matching view and Classifier 2 corresponds to the named entity view. When comparing single classifiers using either keyword matching or named entity features, SVM outperforms all other classifiers with an accuracy of 75.3%. When linearly combining all features using the seed data set, the Bayesian network yields a better result of 77.6%. The Bayesian network co-training outperforms the linear kernel SVM[12] co-training algorithm by 12.4% in accuracy, and it is also 55 times faster than SVM co-training algorithm in terms of average runtime. The co-trained result also has an absolute improvement of 8% over a Bayesian network linear combination classifier (Table 1: Combination).

Note that in all of our co-training experiments, the performance of the Bayesian network co-

[7] Under the same parameter settings, accuracy was reduced by 1% when training only with high-confidence instances (posteriors > 0.9) or low-confidence instances (posteriors < 0.75).

[8] In our experiments, we use the K2 hill-climbing search strategy to estimate network structure with the standard Bayesian scoring metric (Cooper and Herskovits, 1992).

[9] Conditional probabilities are obtained using the simple estimation strategy ($\alpha = 0.5$) as implemented in the Weka machine learning toolkit (Witten and Frank, 2000).

[10] We use $k = 50$ iterations and $v = 500$ sentences per iteration

[11] Here we restrict $k * v = 200,000$ sentences

[12] We have also experimented with polynomial and RBF kernels, but performance was worse than the linear kernel.

Systems		Acc.	N-Prec.	N-Recall	N-F1	P-Prec.	P-Recall	P-F1
Chance		50	–	–	–	–	–	–
Classifier 1	NB	72.39	0.668	0.892	0.764	0.837	0.556	0.668
	SVM	75.29	0.747	0.764	0.756	0.759	0.741	0.750
	Bnet	75.10	0.683	**0.938**	0.790	0.901	0.564	0.694
Classifier 2	NB	70.85	0.706	0.714	0.710	0.711	0.703	0.707
	SVM	71.04	0.709	0.714	0.712	0.712	0.707	0.709
	Bnet	71.04	0.709	0.714	0.712	0.712	0.707	0.709
Combination	NB	75.48	0.777	0.714	0.744	0.736	0.795	0.764
	SVM	73.94	0.752	0.714	0.733	0.728	0.764	0.746
	Bnet	77.61	**0.815**	0.714	0.761	0.746	**0.838**	0.789
Co-training	NB	75.14	0.772	0.714	0.742	0.734	0.788	0.760
	SVM	72.97	0.737	0.714	0.725	0.723	0.745	0.734
	Bnet	**83.98**	0.780	**0.947**	**0.855**	**0.933**	0.733	**0.821**

Table 1: Comparing with baseline systems (NB: naive Bayes. SVM: linear kernel support vector machines. Bnet: Bayesian network. N-: the class of sentences unrelated to the event query P-: the class of event-related sentences). The best result in each column is indicated in boldface.

trained classifier 1 (Table 1: Co-training with $k = 50andv = 500$ improves performance substantially over the Bayesian network classifier using all features (Table 1: Combination). The performance of co-trained Classifier 2, not shown in the table[13], does not improve over the linear feature combination. The results suggest that when the keyword view is augmented with Bayesian network co-trained unlabeled data from both views, the co-training approach boosts the results, but the reverse does not hold true. These results are consistent with our previous claim (Section 3.2) that named entity features help the keyword view to identify key information in events. The above results also indicate that although the keyword view and named entity view are not independent, the Bayesian network co-training approach still produces promising results.

The strong performance of the Bayesian network classifier in these experiments is partly attributable to the fact that both classification views consist of features that are interdependent and therefore correlated to some degree, as the model assumes. The dramatic performance gains of the Bayesian network co-trained classifier when compared to Naive Bayes are therefore somewhat unsurprising; however, the poor performance of the SVM classifier was unexpected.

Although all the basic variants of classifiers using n-gram count features exhibit similar accuracy on the test corpus, we note that the Bayesian classifiers tend to select fewer high-precision event-related sentences resulting in skewed precision/recall numbers. Co-training has the effect of relaxing these models to gradually expand the set of high-precision event sentences (resulting in large gains in P-Recall and N-Precision) using the implicit information gained through new training examples classified by event features. No improvement is observed when co-training for classifiers that start out with balanced precision and recall, i.e., the SVM and all event feature classifiers. We therefore conjecture that this phenomenon of high-precision event sentence extraction is especially helpful in co-training for this task.

We aim to further analyze the effect of classifier choice in future work. For all of the following experiments in this section, we only evaluate the performance of the Bayesian network classifier.

4.2 Trade-off between Co-training Parameters

In this experiment, we evaluate the trade-off between the number of iterations and leap size in our co-training setting given a fixed amount of unlabeled data. A total number of 200,000 unlabeled summary sentences (1 year) are involved in this experiment. Fig. 3 shows the results. The horizontal axis consists of pairs of k and v. We first

[13] Table 1 only shows the best result for co-training.

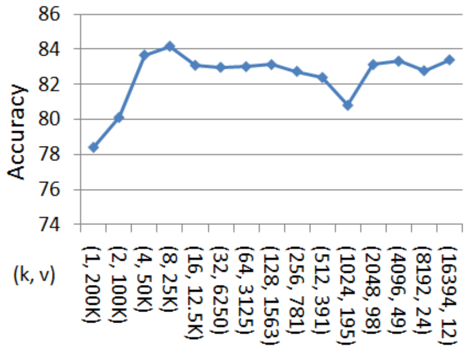

Figure 3: Trade-off between co-training parameters k and v with fixed amount of data

start with a very small ratio of k vs v ($k = 1$, $v = 200,000$), then we increase k exponentially such that $k * v$ is still equal to 200,000. We see that when $k = 8$ and $v = 25,000$, the system's performance reaches a peak accuracy of 84.2%. In addition, accuracy increases 6% from $k = 1$ to $k = 8$ which suggests that our co-training algorithm needs at least 5-10 iterations to achieve satisfactory results. We observe that there is a sudden drop when $k = 1024$ and $v = 195$; we explain this in section 4.4 and 4.3 when we test the impact of data diversity.

4.3 Influence of Leap Size

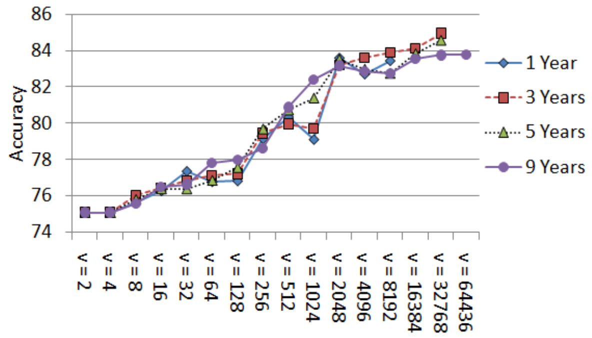

Figure 4: Performance varying leap size v with a fixed number of iterations $k = 10$

We evaluate the co-training leap size parameter v by fixing the number of iterations k to 10 and also experiment with the diversity of data by using unlabeled data drawn from different years. Fig. 4 shows that the system's performance improves significantly when using higher numbers of v. The best result is 85% accuracy when $v = 32,768$, compared to 75.1% accuracy obtained when $v = 2$. We also notice that using more diverse data from different years helps stabilize system performance and reduce the oscillation in the plot. In terms of efficiency, when using all 9 years of data, the corresponding runtime for v = 4 and v = 32768 are 15 minutes and 20 minutes, which indicates that increasing leap size is an efficient approach. Fig. 6 shows the comprehensive runtime costs from v = 2 to v = 2048. The runtime for varying v is almost a straight line, showing very little added cost as v increases.

4.4 Influence of the Iteration Parameter

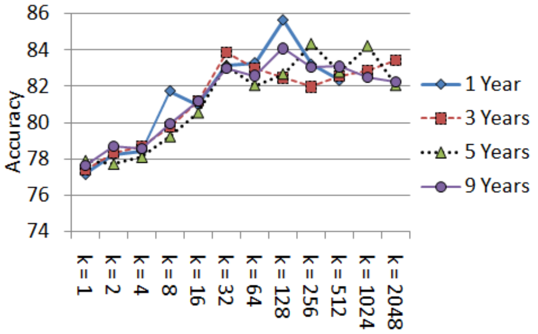

Figure 5: Performance varying the number of iterations k with fixed leap size $v = 500$

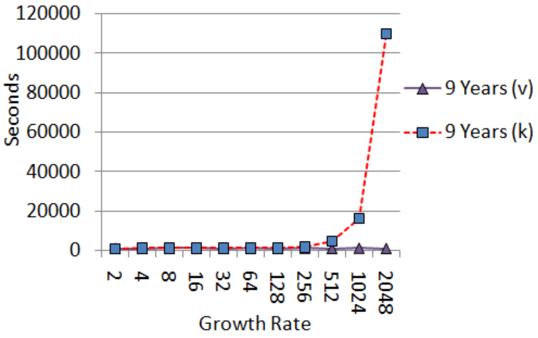

Figure 6: Average runtime costs when varying either k or v separately (and keeping the other fixed)

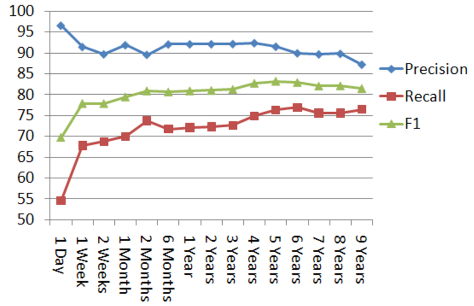

Figure 7: Performance varying the size and temporal range of the unlabeled dataset

We fix the leap size v to 500 and vary the number of iterations k to test its impact on the accuracy and efficiency of our system. We notice (Fig. 5) that when increasing the number of iterations from k = 1 to 32, our system performance improves significantly. After 32 iterations, the performance reaches a stable state. The best performance of 85.6% accuracy is obtained when $k = 128$; however, the cost of system runtime grows exponentially as we increase the parameter k. For example, we note that the runtimes corresponding to k = {256, 512, 1024, 2048} are 28 minutes, 76 minutes, 270 minutes and 1829 minutes. Clearly, compared with the runtimes obtained by varying v, increasing k is not a good option. We also test impact of data diversity by using data from different years. Although using one year of data yields the best result, its overall performance is less stable than using 3, 5, or 9 years of input unlabeled data.

4.5 Influence of Size of Unlabeled Dataset

Experiments 4.3 and 4.4 show that increasing leap size v is an efficient method for introducing more data in co-training. Now, given different amounts of unlabeled news summary data from meaningful temporal time frames, we fix $k = 10$ and dynamically choose the best possible v to represent the amount of input data under consideration. Fig. 7 shows the precision and recall of the event-relevant class when co-training is performed using input unlabeled data of varying sizes (and temporal ranges). We observe that increasing the dataset size (and therefore drawing from older summaries) yields a dramatic improvement in recall accompanied by a slight decline in precision. Correspondingly, F1 generally increases when more data is added. The difference between minimum and maximum recall over the entire range is as high as 22% while precision drops by about 9%. These results suggest that introducing more unlabeled data in our co-training framework has the biggest impact on recall without sacrificing much precision.

5 Discussion

Our results show that an approach that only selects sentences with words that match the input query will not work well. As keyword matching might introduce a fair amount of noise, e.g. matched prepositional phrases that do not indicate the importance of the sentence, co-training together with a named entity based classifier can help reduce these errors. Thus when testing on unseen data, significantly better recall can be obtained via co-training. Our experiments also show that our co-training material, the nine years of news summary documents, is a reliable and effective source to augment the seed dataset, with minimum introduction of new noise or extra overhead.

The experiments show that increasing either the number of co-training iterations k or leap size v leads to improved performance; however, adding unlabeled data by increasing v adds much lower runtime overhead by avoiding repeated training iterations. The parameter trade-off experiment in section 4.2 indicates that beyond a minimum number of iterations k, unlabeled data can largely be introduced through v. This is useful in practice to maintain consistent runtime performance.

Despite the strong relative gains with co-training, the limited feature sets employed in both views (keyword and named entity) do not allow us to capture event-relevant sentences which don't share words with the event query. Although the relative gains offered by co-training are clear, we wonder if the requirement of agreement in annotation led to a corpus in which sentences that were labeled relevant tended to share keywords with the event query. Although not evident from our experiments, we nevertheless assume that query synonyms are likely occurrences in event-relevant sentences, as are sub-events which entail larger events and vice versa. Therefore, a consideration of semantic and ontological features would be a logical next step for further investigation of this task and may offer interesting new views through which co-training can be utilized.

6 Conclusion

In this paper, we aim at detecting event-relevant sentences in a news corpus given an event query. In contrast to previous work that needs expensive annotation for query-answer pairs, we use unlabeled news summaries that are readily available in large quantities online and design an efficient semi-supervised learning strategy for event identification based on co-training with Bayesian network classifiers. An analysis of different parameters in the co-training approach shows that increasing leap size is a better way to gain accuracy improvements than increase in number of iterations given efficiency costs. Our findings are applicable to question answering, summarization, and text mining domains where selection of event-relevant sentences is often critical and large-scale human annotated data is not always available.

References

Bejan, Cosmin Adrian. 2008. Unsupervised Discovery of Event Scenarios from Texts. *Proceedings of the 21st Florida Artificial Intelligence Research Society International Conference (FLAIRS'08), Applied Natural Language Processing track*, Coconut Grove, FL, USA, May 2008.

Bergsma, Shane and Yarowsky, David and Church, Kenneth. 2011. Using Large Monolingual and Bilingual Corpora to Improve Coordination Disambiguation. *ACL-HLT*.

Chambers, Nathaniel and Jurafsky, Dan. 2009. Unsupervised learning of narrative schemas and their participants. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2*, Suntec, Singapore, 2009.

Chirita, Paul - Alexandru and Firan, Claudiu S. and Nejdl, Wolfgang 2007. Personalized query expansion for the web. *SIGIR '07*.

Cooper, G. and Herskovits, E.. 1992. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning* , 9: 309-347, 1992.

Bethard, Steven and Martin, James H.. 2006. Identification of event mentions and their semantic class. *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 146-154.

Blum, Avrim and Mitchell, Tom. 1998. Combining labeled and unlabeled data with co-training. *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100.

Cortes, Corinna and Vapnik, V.. 1995. Support-Vector Networks. *Machine Learning*, 20.

Du, Jun and Ling, Charles X. and Zhou, Zhi-Hua. 2010. When Does Co-Training Work in Real Data? *IEEE TKDE*.

Filatova, Elena and Hatzivassiloglou, Vasileios. 2003. Domain-Independent Detection, Extraction, and Labeling of Atomic Events. *Proceedings of RANLP'03*, 2003.

Filatova, Elena and Hatzivassiloglou, Vasileios. 2004. Event-Based Extractive Summarization. *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, Barcelona, Spain, July 2004.

Finkel, Jenny Rose and Grenager, Trond and Manning, Christopher. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 363-370.

Fisichella, Marco and Stewart, Avaré and Denecke, Kerstin and Nejdl, Wolfgang; 2010. Unsupervised public health event detection for epidemic intelligence. *Proceedings of the 19th ACM international conference on Information and knowledge management (CIKM 2010)*, pages 1881–1884, Toronto, ON, Canada.

Krogel, Marc-A and Scheffer, Tobias. 2004. Multi-Relational Learning, Text Mining, and Semi-Supervised Learning for Functional Genomics. *Machine Learning 57*, pages 61-81.

Parton, Kristen and McKeown, Kathleen R. and Allan, James and Henestroza, Enrique.. 2008. Simultaneous Multilingual Search for Translingual Information Retrieval. *Proceedings of CIKM*.

Pustejovsky, James and Hanks, Patrick and Saur, Roser and See, Andrew and Gaizauskas, Robert and Setzer, Andrea and Radev, Dragomir and Sundheim, Beth and Day, David and Ferro, Lisa and Lazo, Marcia. 2003. The TIMEBANK Corpus. *Proc. of Corpus Linguistics*.

Li, Shoushan and Wang, Zhongqing and Zhou, Guodong and Lee, Sophia Yat Mei. 2011. Semi-supervised Learning for Imbalanced Sentiment Classification. *Proceedings of IJCAI-2011*.

Mani, Inderjeet and Schiffman, Barry and Zhang, Jianping. 2003. Inferring Temporal Ordering of Events in News. *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, 2003.

Manshadi, Mehdi and Swanson, Reid and Gordon, Andrew S.. 2008. Learning a Probabilistic Model of Event Sequences From Internet Weblog Stories. *Proceedings of the 21st Florida Artificial Intelligence Research Society International Conference (FLAIRS'08), Applied Natural Language Processing track*, 2008.

Marge, Matthew and Banerjee, Satanjeev and Rudnicky, Alexander I. 2010. Using the Amazon Mechanical Turk for Transcription of Spoken Language. *Proceedings of ICASSP*, 2010.

Siegel, Eric V. and McKeown, Kathleen R. 2000. Learning methods to combine linguistic indicators: improving aspectual classification and revealing linguistic insights. *Comput. Linguist.*, Dec. 2000.

Mihalcea, Rada. 2004. Co-training and self-training for word sense disambiguation. *Proceedings of CONLL - 04*, 2004.

Radev, Dragomir R. and Blair-Goldensohn, Sasha and Zhang, Zhu and Raghavan, Revathi Sundara. 2001. NewsInEssence: a system for domain-independent, real-time news clustering and multi-document summarization. *Proceedings of the first international conference on Human language technology research*, 2001.

Snow, Rion and O'Connor, Brendan and Jurafsky, Daniel and Ng, Andrew Y.. 2008. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2008.

Rosenthal, Sara and Lipovsky, William and McKeown, Kathleen and Thadani, Kapil and Andreas, Jacob. 2010. Towards Semi-Automated Annotation for Prepositional Phrase Attachment. *LREC*, 2010.

Spitters, M. and Kraaij, W.. 2002. Unsupervised event clustering in multilingual news streams. *Proceedings of the LREC2002 Workshop on Event Modeling for Multilingual Document Linking*, pp. 4246.

Wan, Xiaojun. 2009. Co-training for cross-lingual sentiment classification. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp. 235–243.

Witten, I.H. and Frank, E.. 2000. Data mining: Practical machine learning tools and techniques with Java implementations. *Morgan Kaufmann*, 2000.

Xu, Jinxi and Croft, W. Bruce. 1996. Query expansion using local and global document analysis. *SIGIR '96*.

Yu, Ning and Kübler, Sandra. 2011. Filling the Gap: Semi-Supervised Learning for Opinion Detection Across Domains. *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*.