Semantic Text Classification of Emergent Disease Reports

Yi Zhang and Bing Liu

Department of Computer Science, University of Illinois at Chicago,
851 S. Morgan Street, Chicago IL 60607, USA
{yzhang3, liub}@cs.uic.edu

Abstract. Traditional text classification studied in the information retrieval and machine learning literature is mainly based on topics. That is, each class or category represents a particular topic, e.g., sports, politics or sciences. However, many real-world problems require more refined classification based on some *semantic perspectives*. For example, in a set of documents about a disease, some documents may report outbreaks of the disease, some may describe how to cure the disease, some may discuss how to prevent the disease, etc. To classify text at this semantic level, the traditional bag-of-words model is no longer sufficient. In this paper, we study semantic text classification of disease reporting. We show that both keywords and sentence semantic features are very useful for the classification. Our experimental results demonstrated that this integrated approach is highly effective.

Keywords: Semantic text classification.

1 Introduction

In traditional topic-based text classification, the bag-of-words representation of text documents is often sufficient because a topic can usually be characterized by a set of topic-specific keywords [3, 20, 22]. However, for semantic text classification, the unigram or n-gram representation is no longer sufficient because the texts from different classes may be on the same large topic. For example, in a set of documents about a particular disease, some documents may report outbreaks of the disease, some may describe how to cure the disease, and yet some may discuss how to prevent the disease. To classify texts at such level, the system needs to capture some semantic characteristics of the texts in order to perform more accurate classification.

In this paper, we propose to integrate the bag-of-words scheme and semantic features extracted from texts for classification. As a case study, we investigate the disease reporting domain. We want to classify sentences that report disease outbreaks, and sentences that do not. For example, the following sentence reports a possible disease outbreak "*the district hospital reported today that 10 people were diagnosed with cholera this morning*". However, the following sentence does not report an outbreak, "*the district hospital reported today that they have successfully tested a new cholera treatment procedure*". Both sentences are on the topic of cholera. However, they are entirely different semantically. The problem is how to separate sentences based on the required semantic categories, i.e., reporting a possible outbreak or not in this case. This classification task is an important application in its own right. The work is supported by an environmental agency. We note that sentences rather than documents are used in this work because a document contains a large number of sentences and the sentences have quite different semantic meanings. For example, a piece of disease outbreak news may contain many pieces of other related information, e.g., symptoms, treatment, vaccine, and past disease history.

This paper shows that both the words used in sentences and the sentence semantic characteristics are important. Our experimental results confirm that this integrated approach produces much more accurate classifiers than each of them alone.

2 On Semantic Text Classification

The setting of semantic text classification is the same as traditional topic-based text classification. We have a set of documents D and each document $d_i \in D$ is labeled with a class $c_j \in C$, where C is a set of known classes. A supervised learning algorithm is applied to build a classification model. However, semantic text classification usually has more refined categories or classes. Different classes are hard to be separated based on bag-of-words or n-grams alone. Semantic information is required. For example, the sentence, "*the district hospital reported that 10 people were diagnosed with cholera early today*", reports a possible cholera outbreak. It is easy to observe that the words "reported" and "diagnosed" are indicative of an outbreak. The times, "today" and "this morning", indicate a new event. Using the words alone, however, is insufficient as the following sentence illustrates: "*10 people from the district hospital submitted a report early today on cholera diagnosis*." This sentence uses very similar words, but has a completely different semantic meaning, i.e., it does not report a disease outbreak at all.

In this paper, we define *semantic information* as any information extracted from the text that is not based on keywords or n-grams. Clearly, there are multiple levels of semantic information. At the highest level, it is the full understanding of the text, which is still not possible with the current technology. At lower levels, we have features with different amounts of semantic contents, which can be extracted from sentences based on the current NLP techniques. The exact features used in this work will be discussed in the next section.

An interesting question is whether the bag-of-words representation is still useful in semantic text classification. We believe that the answer is yes for two reasons:

1. To express a particular semantic meaning, certain specific keywords are still more likely to be used, although the same words can be used to express other information but with less frequency.
2. Semantic feature extraction is still not perfect. There may be many errors. Keywords can help offset some of these errors.

Fig. 1. illustrates the difference between traditional text classification and semantic text classification as described in this work. Note that we do not make a difference of the types of classes or texts used in a classification task because classification of any type of categories may be assisted by some level of semantic information.

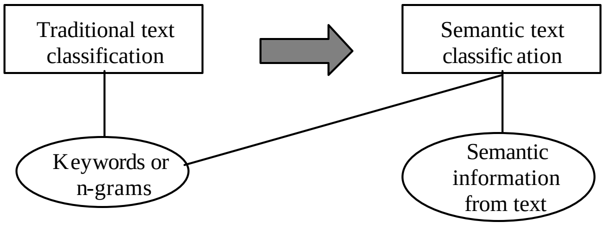
Traditional text classification
Semantic text classific ation
Keywords or n-grams
Semantic information from text

Fig. 1. Traditional text classification and semantic text classification.

3 Important Semantic Features

Our aim is to classify sentences that report possible disease outbreaks and those that do not, which is a classification problem. We will use a supervised machine learning algorithm, e.g., naïve Bayesian (NB) or support vector machines (SVM). Thus, we only need to design and construct features. As we mentioned above, we use both keywords and semantic features. Keyword features are obtained in the same way as in traditional text classification. Here, we only focus on semantic features.

3.1 Noun Phrase Containing a Disease Word

Center word of a noun phrase: Noun phrase is the basic building brick of a sentence's structure. In the traditional bag-of-words approach, all words in a noun phrase are treated equally. However, the center word in a noun phrase has a more direct influence on the overall semantic meaning than any other word in the noun phrase. For example, in the sentence, "*the district hospital reported today that their new cholera treatment procedure had been very successful*", the center word of the noun phrase "*their new cholera treatment procedure*" is "*procedure*". While in "*10 cases of cholera have been reported early today*", the center word of the noun phrase "*10 cases of cholera*" is "*cases*". Thus, although the disease name "cholera" appeared in both sentences, it has different center words, which lead to different semantic meanings. Note that we only use noun phrases that contain a disease word because such phrases are more likely to be relevant to our classification task.

Negation modifier and determiner word: Other important features in a noun phrase include negative modifiers such as "no", and determiner words such as "every" and "a". Their importance can be illustrated by the following examples: "*No case of cholera has been found yesterday*" indicates no disease found, and "*For every case of mad cow disease in Switzerland, 100 animals may carry the infection silently*" gives a study result on the disease rather than a specific case.

3.2 Verb Phrase

Verb and adjective: The verb serves as the main skeleton of a sentence and thus an important feature. Sometimes, a verb is too common to have a specific meaning. In that case, the adjective word after the verb becomes important. For example, the verbs "is" and "become" are not specific, but "is ill" and "become sick" are.

Tense: Another characteristic of a verb phrase is the verb's tense, which is also important in the semantic meaning because tense may show the time or the subjunctive mood. Past perfect tense usually means something happened in the long past. For example, "*West Nile Virus had plagued US*" refers to an old disease outbreak. Subjunctive mood expressed by the tense of past-future perfect is often used for conjectures or assumptions. For example, "*a bird flu outbreak could have killed millions of people*" is a conjecture of the disease's impact rather than a report.

Auxiliary word: Subjunctive mood can also be expressed by a verb's auxiliary words such as "can" and "may". Again, it shows a hypothetical case instead of a fact.

Verb phrase being an if-whether clause: Using word "if" or "whether" is yet another way to express the subjunctive mood. Similarly, it does not describe a fact.

Negation word of verb phrase: A verb with negation modifiers usually has the opposite meaning to the verb alone. Examples of such modifiers are "not", "rarely", "seldom", "never", and so on.

Verb phrase being an adjective clause: If a verb phrase appears in an adjective clause of a sentence, it often gives complementary information, not the main content of the sentence. For instance, in the sentence "*the team reports on their investigation of a Canadian farm where an outbreak of pneumonia in pigs began in October 1999,*" the main interest is not in reporting the outbreak, as the outbreak most likely happened in the past and has been noticed before.

Subject and object: The subject word and object word of a verb are also important. As in "*The real name for mad cow disease is Bovine Spongiform Encephalopathy*", the subject "name" and the object "*Bovine Spongiform Encephalopathy*" suggest that the sentence is not about any disease outbreak.

3.3 Dates

For disease outbreak reporting, dates used in sentences are important. If the date appearing in a sentence refers to a long time ago, the sentence is unlikely to report a new outbreak. Although the verb tense can show whether the time is in the past, present, or future, it is ambiguous as it is unclear how far in the past or in the future.

Date information can be expressed in a large number of ways. We focus on the most common ones in this work. Thus, our description below is by no means complete, but is quite sufficient for our data. A piece of date information is usually expressed by a prep word (implied if it is missing or omitted) followed by a *date phrase*. We call the date expressed in the text as the *expressed date*, and the date of the context as the *context date* (e.g., the date when a news report was published)

Prep word: A prep word decides the relationship between an *expressed date* and the date phrase that follows. We summarize the prep words and the corresponding relationships in Table 1. If a prep word is omitted, in most cases it's the same as the first relationship in the table. For example, "*The alert was given <u>last Tuesday</u>*".
Adjective and adverb: Some adjectives and adverbs may also be associated with

dates, e.g., "ago" as in "three months ago", "last" as in "last month", etc. Grammar rules related to them will be given below.

Relationship	Prep word	Example
expressed date is the date phrase	in, at, on, during	on Monday
expressed date is before the date phrase	before	before winter
expressed date is after the date phrase	after	after May 1, 2006
expressed date ends within the date phrase	in, within	in two days
expressed date ends by the date phrase	by, as of, until, till, no later than	by today
expressed date spans the two date phrases	Between … and… from…to…	between January and February
expressed date starts from the date phrase	since	since last year

Table 1. Prep words and relationships between *expressed dates* and date phrases.

In general, a *date phrase* expresses either an *absolute date* or a *relative date*. We will not discuss time in this paper as the date information is sufficient for our application task. The ways to express a specific time are not as diverse as those for date and can be dealt with in a similar way.

Absolute date: As its name suggests, an absolute date expresses a specific date without ambiguity regardless when it is seen. There are two main types:

- *Historic period*: It is a time period in history. Its duration is usually very long, and it has a specific name, e.g., "Stone Age".
- *Formal date*: It specifies an absolute time period in quantitative terms that can be:
  a century (e.g., "18th century"),
  a decade (e.g., "1980's"),
  a year (e.g., 1998),
  a season (e.g., summer of 2007),
  a month (e.g., May 1998),
  a day (e.g., October 22, 2005),
  a time period of a specific day (e.g., morning of Mar 22, 2005),
  …

Relative date: The absolute date of a *relative date* can only be determined based on the *context date*.

- *Recurrent named date*: Such a relative date occurs repetitively, e.g., annual festival, season, month of year, day of month, day of week, etc. For example, "May 22" refers to the date in the year determined by context. Other examples include, "last Christmas", "next morning", and "this Thursday". A restrictive modifier is usually mandatory, although sometimes it is omitted based on convention. For example, in "*an outbreak was reported on Monday*", "Monday" usually refers to "the *past* Monday".

- *Other named dates*: Such dates include “today” and “tomorrow” or *special words* (e.g., “now” and “recently”) that are dedicated to some relative dates. No restrictive modifier such as “next” or “last” is needed before them.
- *Number-unit*: This is also popularly used in date phrases, e.g., “three months” in “three months ago” and “ten years” in “past ten years”. Similar to a recurrent date, a modifier is also required for this type, e.g., “ago” and “past”.

A date phrase may have a refiner, such as “*early* 2007” and “the *end* of last month”. Now we give a formal definition of date phrases in Backus–Naur form. Due to space limitations, some rules use suspension points in place of similar entries.

```
  <DatePhrase> ::= [<Refiner>]<FormalDate>
  | [<Refiner>]<HistoricPeriod> | [<Refiner>]<FormalPeriod>
  | [<Refiner>]<Modifer><DateName> | <SpecialWord>
  | [<Refiner>]<Modifier>[<Number>]<Date Unit>
  | [<Number>]<Date Unit>[<PostModifier>] | [<Refiner>]<SpecialDay>
  <Refiner> ::= fiscal | late | early | end of | beginning of
| middle of | mid | ……
  <Modifier> ::= last | previous | next | coming | past|……
  <PostModifier> ::= ago | later | early | ……
  <DateUnit> ::= century | decade | year | season | quarter
  | month | week | day | hour | minute | second
  <SpecialDay> ::= today | tomorrow | the day before yesterday | ……
  <SpecialWord> ::= now | recently | …
  <FormalDate> ::= [<Month>] <Year> | [<Season>] <Year>
  | [<Festival>] <Year> | [<DayofWeek>]<Month>[/]<Day>[/][<Year>]
  <HistoricPeriod> ::= stone age | ……
  <FormalPeriod> ::= <OrdinalNumber> century | <year>[’]s | ……
  <DateName> ::= <Festival> | <DayOfWeek> [<TimeOfDay>]
  | <Season> | <MonthOfYear>
  <Month> ::= <Digit><Digit> | <MonthOfYear>
  <Day> ::= 1 | 1st | …… | 31 | 31st
  <Year> ::= <Digit><Digit><Digit><Digit>
  <Season> ::= spring | summer | fall | autumn | winer
  <MonthOfYear> ::= jan | january | feb | feburary | ……
  <DayOfWeek> ::= mon | monday | tue | tuesday | ……
  <TimeOfDay> ::= morning | noon | afternoon | evening | ……
  <Festival> ::= Christmas [eve] | [post] Christmas |……
  <OrdinalNumber> ::= 1st | first | 2nd | second | ……
  <Number> ::= <Digit>+ | one | two | ……
  <Digit> ::= 0 | 1 | …… | 9
```

Most date phrases can be used after any prep word, with some exceptions, e.g., “three days ago” is usually not used with “during”.

4 Feature Extraction

Section 3 introduced several features that are important for semantic classification of disease sentences. Now we describe how to extract these features from a sentence.

4.1 Named entity recognition

Named entities representing disease names and dates are essential for feature extraction because most features described in Section 3 can only be found based on correct recognition of the corresponding named entities, i.e., disease names or dates. Thus, recognizing named entities is a necessary step. The named entity recognition system that we use is given in Section 4.4. Note that a typical named entity recognizer also recognizes locations, person names, organization names, etc, but they are not needed in this work.

4.2 Dependency tree

A dependency tree describes the syntactical and semantic relationships between pair of words in a sentence. Fig. 2. shows an example dependency tree, generated from the sentence "*Danish health authorities on Friday confirmed the Scandinavian country's fourth case of mad cow disease*". Preposition words and the word "'s" are put on the edge to save space. In a dependency tree, arrows point from a parent node to a set of children nodes that the parent node governs. The dependency relationship between a parent node and a child varies. For example, nodes "authorities" and "confirmed" have a subject-and-verb relationship while "fourth" and "case" have a noun-noun-modifier relationship. Note that "mad cow disease" is a named entity, so it is represented by a single node in spite of that it has three literal words.

Because the natural language is very flexible, there are many ways to express the same idea. For example, the above example can be rephrased in a passive sentence "*The fourth case of mad cow disease in the Scandinavian country has been confirmed by Danish health authorities*". Though the words are in a different order, this sentence will actually generate almost identical dependency tree as the previous sentence does.

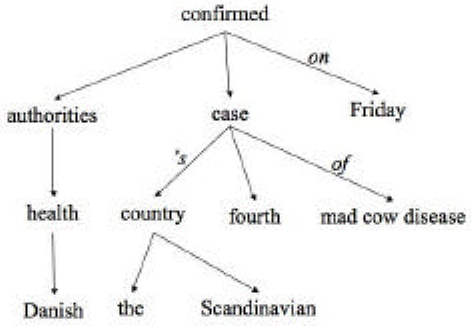

Fig. 2. An example of a dependency tree

4.3 Feature extraction and construction

After a sentence's named entities have been recognized and the dependency tree has been built, features are extracted in the following way:

We start from a named entity of any infectious disease, and find the noun phrase that contains the disease. The center word of a noun phrase is the highest node in the noun phrase's dependency tree. Negation modifier and determiner words can be found among children nodes of the center word.

The verb can be determined easily as it is always the nearest ancestral verb node of the noun phrase. If there is an adjective node between the verb and the noun phrase, it is taken as the adjective word feature. The tense of a verb phrase does not depend solely on the form of the center verb. The forms of auxiliary words such as "*do*", "*have*" and "*be*" count as well. For example, in "*an outbreak of cholera has been reported*", although the center verb "*reported*" is of the preterit form, it is not past tense because of the auxiliary word "*has*". All auxiliary words are children nodes of the center word. So are negation modifiers, and subject/object words.

To get other features of a verb phrase, we need to check sibling or parent nodes of the verb node in the dependency tree. If a verb phrase is an adjective clause, the verb node normally has a sibling node of "wh-" word and a relationship of complementary to its parent node. Here is an example, "*the team reports on their investigation of a Canadian farm where an outbreak of pneumonia in pigs began in October 1999*". If a verb phrase is an if-whether clause, there will be a sibling node of "if" or "whether". By scanning the verb's sibling nodes, these features can all be found easily.

For the date feature, it can be recognized using the definitions given in Section 3.3. However, it is not trivial to normalize dates so that they will be comparable to each other. One solution is to translate all dates into absolute ones, and construct features that include a date's year, month, and day. But for nonspecific dates, such as "*during the last decade*", accurate translation is impossible. In this application, we are only interested in recent disease outbreaks, so we generally treat a relative date in the scope of current year as "recent", i.e., things happened in sometime last year and before would be "old". Any date after the report date is considered a "future" date. Thus, the date feature has three possible values, "recent", "old" and "future".

In some sentences, there are multiple disease names in one sentence, and then additional features are created as long as they correspond to different noun phrases.

4.4 Implementation

We use the English parser MINIPAR [12] for dependency tree generation and for named entity recognition. In order to recognize infectious diseases, we supplemented the standard MINIPAR database with infectious disease names extracted from Centers for Disease Control and Prevention (http://www.cdc.gov/ncidod/). Another modification is to recognize some date phrases such as "last week", which is not recognized by MINPAR. The feature construction algorithm, which is implemented in Perl, then reads the generated dependency trees and outputs features.

5 Experiments

This section evaluates the proposed technique. We discuss the experimental data, evaluation settings and the results in turn.

Experimental data: Our corpus consists of sentences related to infectious diseases. Some of the sentences are emergent disease reports (EDR), and others are not (non-EDR) but still contain the disease names. The sentences are extracted from disease report documents from ProMED-mail (http://www.promedmail.org). We labeled the sentences into two classes: EDR (Emergent Disease Report) and nonEDR. The data set has 1660 nonEDR sentences and 682 EDR sentences.

Experimental settings: Two popular supervised learning algorithms are used to build models, Support Vector Machines (SVM) and naïve Bayesian (NB). Both algorithms are provided in the latest version of the Rainbow package [13], which is used in our experiments. Different types of features are employed and compared:

– **sentence**: only bag-of-words representation with *i*-grams: 1-gram, 2-gram, 3-gram and 4-gram.

– **s-features**: semantic features (including the date feature).

– **s-features+sentence**: semantic features and bag-of-words features in a sentence are combined.

To ensure reliable results, we run each technique 10 times. In each run, 80% of the data (randomly selected) is used for training and 20% of the data is used for testing. The results are then averaged and reported below. The evaluation measure is F-score on EDR sentences. F-score is the harmonic mean of precision (*p*) and recall (*r*), i.e., $F = 2pr/(p+r)$, which is commonly used in text classification.

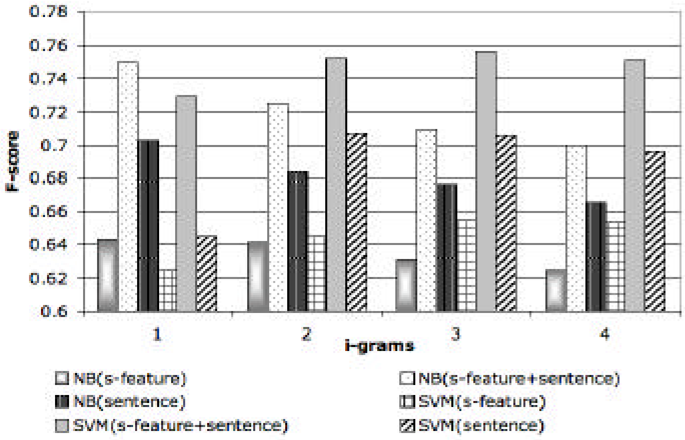

Fig. 3. Experimental results

Results: Fig. 3 shows the average F-scores of all methods. We observe that semantic features (denoted by s-features) are very helpful. Both SVM and NB produce much better results when sentences and semantic features are both used. SVM (s-features+sentence) with 3-gram for sentences gives the best F-score and it also performs the best for 2-gram and 4-gram, except for 1-gram, in which NB (s-features+sentence) is better.

We also single out the date feature to see how it effects classification as intuitively the disease reporting dates are important for EDR sentences.

The date feature is indeed helpful (Fig. 4). For NB, the F-scores with date features are always better than without date features. For SVM, the results are also better for 1-gram and 2-gram. All the results here use both s-features and sentences. Note that there are 271 nonEDR and 240 EDR sentences with "recent" for the date feature. Thus, the classification cannot be done trivially using dates alone.

Since the date feature has shown its importance, it will probably help more if the weight of the date feature is increased. We thus increase its weight. Multiplying each date feature by 3 gives the best results. Fig. 4 shows that "with triple date feature" (the other settings remain the same) gives better F-scores for both NB and SVM in almost all cases. NB with 1-gram produces the best result. Due to this success, we also tried to increase the weight s of all s-features ("with all features doubled") but without improvements (Fig. 4).

In summary, we can conclude that combining bag-of-words and semantic features indeed improves classification. The date feature is also very helpful.

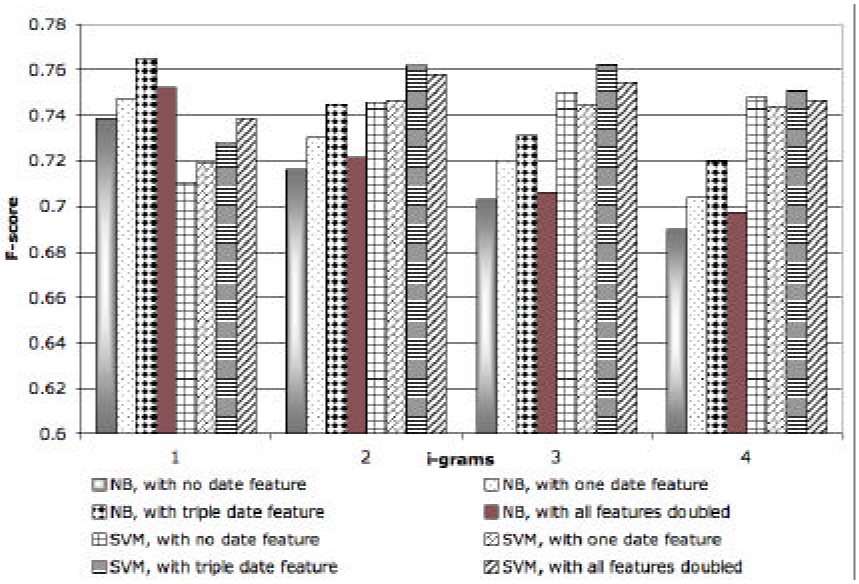

Fig. 4. Experimental results: no date feature

6 Related work

There are several works on using linguistic information for text classification. Most of them are still based on the idea of carefully choosing additional keywords or phrases.

Noun phrases have been used in the work of Furnkranz et al. [4], and higher precision but lower recall were reported. Aizawa [1] incorporated terms features extracted based on part-of-speech tags. Sahlgren and Coster [19] improved the performance of SVM in text classification by using bag-of-concepts. Moschitti and Basili [15] investigated adding complex nominals as features. Hulth and Megyesi [8] reported classification results based on several keyword extraction methods, i.e., 1-

grams, 2grams, and 3grams, noun phrase chunks, and frequently occurring POS patterns. They showed that the combination of features gives the best result.

Ko et al. [9] assigned feature weights based on the importance of each sentence determined by a text summarization system, and observed an improvement in classification performance. Mihalcea and Hassan [14] also used automatic extractive summarization in text classification, while their approach is to integrate a graph-based method for automatic summarization with a text classifier. Li et al. [11] took advantage of existing summaries of texts as well.

Dependency trees were also used in classification before. Kudo and Matsumoto [10] showed that sub-trees of dependency trees are helpful in classification, but they also found that using n-gram produces comparable results.

Our work is related but also different from these existing text classification works in several ways. First, they still focus on classifying whole text documents using benchmark data sets such as 20-newsgroups and Reuters-21578, which are typical topic-based classification data sets. Keywords and possibly noun phrases are quite good representation of the topics [20]. However, our task focuses on sentence level classification, which requires more semantic information, i.e. more delicate features from sentences based on the dependency tree (e.g., center noun, negation word, determiner, tense, etc). We also extract dates and treat them as features, which to our knowledge have not been done before for text classification.

Sentence level classification is commonly applied in sentiment analysis or opinion mining, where the system determines whether a sentence expresses a positive or a negative opinion [e.g., 2, 6, 7, 16, 17, 18, 21]. They mainly use opinion words (e.g., great, wonderful, bad, and poor) phrases in the process. Our task is quite different from sentiment analysis and thus requires different features for classification. We also use both the bag-of-words representation and semantic features and dates from sentences in classifier building.

In the infectious disease reporting domain, there are some existing proprietary systems, e.g., GPHIN (Global Public Health Intelligence Network, http://www.phac-aspc.gc.ca/) and HealthMap (http://healthmap.org). GPHIN uses a filtering system (unpublished) and human experts to identify potential infectious disease outbreaks from news. Other systems such as HealthMap generally do not discriminate between EDR and nonEDR. Grishman et al. [5] reported a system that extracts disease outbreak information, but their task is information extraction, not text classification.

7 Conclusion

In this paper, we studied the problem of classifying disease reporting at the semantic level. It is shown that both the bag-of-words and semantic features are valuable for the classification task. We identified and extracted a set of important semantic features that can be reliably identified by an existing parser and utilized them in classification. We also investigated the representation and extraction of dates in great details, which we believe will also be useful to other applications. Experimental results demonstrated that the proposed integrated approach significantly outperforms each individual approach alone. In our future work, we plan to investigate in two directions. First, we plan to further improve the classification accuracy of our task.

Second, we hope to study the problem of semantic text classification in general rather than dealing with each specific application.

Acknowledgments. This work is funded by the Great Lakes Protection Fund. We also thank Karl Rockne for helpful discussions.

Reference s

1. A. Aizawa. Linguistic techniques to improve the performance of automatic text categorization. In *NLPRS-01*, 2001.
2. K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: Opinion extraction and s emantic classificat ion of product reviews. In *WWW-03*, 2003.
3. G. Forman: Tackling concept drift by temporal inductive transfer. In *SIGIR-06*, 2006.
4. J. Furnkranz, T. Mitchell, and E. Riloff. A case study using linguistic phrases for text categorization on the WWW. In *AAAI-98 Workshop on Learning for Text Categorization*.
5. R. Grishman, S. Huttunen, and R. Yangarber. Real-Time Event Extraction for Infectious Disease Outbreaks. In *HLT-02*, 2002.
6. V. Hatzivassiloglou and J. Wiebe. Effects of Adjective Orientation and Gradability on Sentence Subjectivity. In *COLING-00*, 2000.
7. M. Hu and B. Liu. 2004. Mining and summarizing customer reviews. In *KDD-04*, 2004.
8. A. Hulth and B. Megyesi. A Study on Automatically Extracted Keywords in Text Categoriz ation. In *ACL-06*, 2006.
9. Y. Ko, J. Park, and J. Seo. Improving text categorization using the importance of sentences. *Info. Proc. and Manag.* 40(1):65–79, 2004.
10. T. Kudo and Y. Matsumoto. 2004. A boosting algorithm for classification of semi-structured text. *EMNLP-2004*.
11. C. Li, J.-R. Wen, and H. Li. Text classification using stochastic keyword generation. In *ICML-03*, 2003.
12. D. Lin and P. Pantel. Discovery of Inference Rules for Question Answering. *Nat. Lang. Eng.*, vol.7-4, 2001.
13. A. McCallum. *Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering*. http://www.cs.cmu.edu/~mccallum/bow, 1996.
14. R. Mihalcea and S. Hassan. Using the essence of texts to improve document classification. In *RANLP-2005*, 2005.
15. A. Moschitti and R. Basili. Complex linguistic features for text classification: A comprehensive study. In *ECIR-04*, 2004.
16. V. Ng, S. Dasgupta and S. M. Niaz Arifin. Examining the Role of Linguistic Knowledge Sources in the Automatic Ident ification and Classification of Reviews. In *ACL-06*, 2006
17. B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *EMNLP-02*, 2002.
18. E. Riloff and J. Wiebe. 2003. Learning extraction patterns for subjective expressions. In *EMNLP-2003*.
19. M. Sahlgren and R. Coster. 2004. Using bag-of-concepts to improve the performance of support vector machines in text categorization. In *COLING 2004*.

20. F. Sebastiani. 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47.
21. P. Turney. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. *ACL-02,* 2002.
22. Y. Yang and X. Liu. A re-examination of text categorization methods. In *SIGIR-99,* 1999.