On Document Splitting in Passage Detection

Nazli Goharian
Information Retrieval Lab
Computer Science Department
Illinois Institute of Technology
Chicago, Illinois, U.S.A
nazli@ir.iit.edu

Saket S.R. Mengle
Information Retrieval Lab
Computer Science Department
Illinois Institute of Technology
Chicago, Illinois, U.S.A
saket@ir.iit.edu

ABSTRACT

Passages can be hidden within a text to circumvent their disallowed transfer. Such release of compartmentalized information is of concern to all corporate and governmental organization. We explore the methodology to detect such hidden passages within a document. A document is divided into passages using various document splitting techniques, and a text classifier is used to categorize such passages. We present a novel document splitting technique called dynamic windowing, which significantly improves precision, recall and F1 measure.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Filtering

General Terms

Algorithm, Experimentation

Keywords

Passage Detection, Text Classification

1. INTRODUCTION

Transferring information outside organizational boundaries is a concern to both commercial and governmental organizations. Such information can be hidden as passages within text. It is not feasible to manually check for such passages within large documents.

Traditionally, text classifiers are used to identify the topic of a document. Text classifiers treat each document as a single classification unit and assign one or more categories to that document. However, a document may contain hidden passages whose contents differ from the assigned category of that document. Though text classifiers work effectively to assign categories to documents, they fail to identify such hidden passages.

Passage retrieval research efforts have addressed approaches to find passages in a document that match a user query, or even an expanded user query such as using relevance feedback. However, the passage retrieval approaches do not identify the passages based on the subject matter, or category of content of such passages. Our focus is on passage detection and not passage retrieval, and thus, we provide a differentiation of the two:

- Passage detection attempts to identify passages related to user specified topics (category), while passage retrieval concerns with passages related to user queries.
- In passage detection, training documents are used to train a classifier on a topic, while passage retrieval is generally not a supervised process.
- In passage detection, the effectiveness of results depends on the accuracy of the text classification model. In passage retrieval, the effectiveness of results depends not only on the engine but also on how the query is formulated by a user.

In our earlier efforts [2] we used a three-phase methodology for hidden passage detection. In the first phase, training documents are used to build a text classification model based on the document terms and apriori known categories of these documents. In the second phase, the documents are divided into passages using well-known document splitting techniques. In the third phase, the text classification model is used to detect the infected documents, i.e., the documents that contain a passage related to a user specified category, which is different than the category of the document.

In [2] we explored the window based and structure based approaches for passage detection. We present a novel document splitting technique for passage detection that defines passages around significant terms. Our results show that our proposed method statistically significantly outperforms the previously introduced document splitting techniques for the task of passage detection in terms of precision, recall and F1.

2. PRIOR WORK

A passage is defined as any sequence of text from a document. As the definition of passage is vague, different types of automatic document splitting techniques exist. [1] demonstrates that *overlapping passage* approach performs significantly better than the other approaches in passage retrieval task.

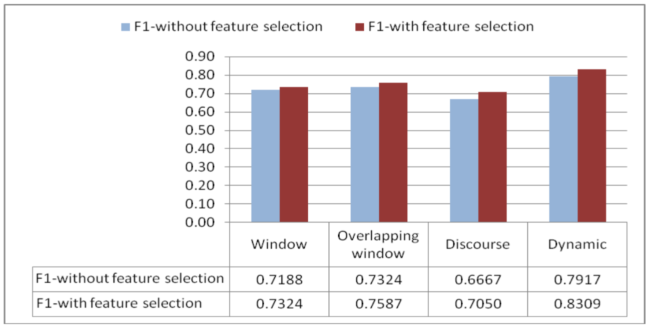
In [2] we explored three known document splitting approaches namely *non-overlapping window* passage approach, *overlapping window* approach and *discourse passage* approach and mapped them to the problem of passage detection. The *non-overlapping window* based passage approach defines a passage as *n* number of words. In *overlapping window* passage approach, a document is divided into passages of evenly sized blocks by overlapping n/2 words from the prior range and n/2 words from the next range. *Discourse passages* are based on logical components such as discourse boundaries such as a sentence.

3. METHODOLOGY

We present a method for document splitting in detection task called *dynamic windowing*. In our earlier efforts that adapted document-splitting techniques from passage retrieval, we did not use the information regarding the category of a document. However, in text classification, feature selection algorithms

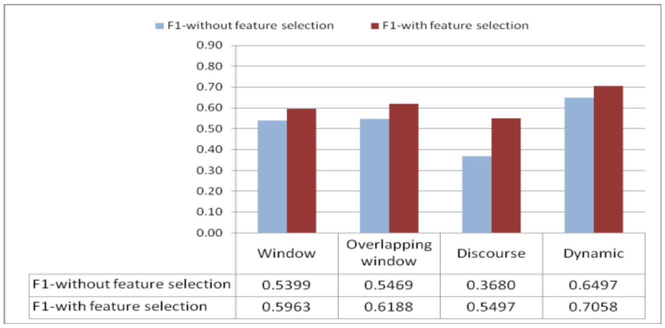
SIGIR'08, July 20–24, 2008, Singapore.

Figure 1. *Passage Detection* results on 20NG



	Window	Overlapping window	Discourse	Dynamic
F1-without feature selection	0.7188	0.7324	0.6667	0.7917
F1-with feature selection	0.7324	0.7587	0.7050	0.8309

Figure 2. *Passage category prediction* results on 20NG



	Window	Overlapping window	Discourse	Dynamic
F1-without feature selection	0.5399	0.5469	0.3680	0.6497
F1-with feature selection	0.5963	0.6188	0.5497	0.7058

assign a weight to each document term to indicate the strength of relevance of a term to a given category. We used Naïve Bayes classifier using *odds ratio* feature selection algorithm and also we used the same classifier (FACT) and feature selection method as used in [2]. A weight called *Ambiguity Measure* (*AM*) [2,3,4] is assigned to each document term based on how ambiguous a term is in respect to a given category C_i. The formulae for calculating AM for a term t_k are given in formula 3.1 and 3.2.

$$AM(t_k, C_i) = \left(\frac{tf(t_k, c_i)}{tf(t_k)} \right) \qquad ..3.1$$

$$AM(t_k) = \max(AM(t_k, C_i)) \qquad ..3.2$$

In our *dynamic windowing* approach passages are defined around terms with higher *AM* weights. We assume that the probability of detecting the correct category of a passage is higher when the passage contains a term with higher weight (i.e. less ambiguous terms). Thus, for a fixed length of the passage that is *n* words long, we define a passage from *n/2-1* terms before a term with higher weight and up to *n/2* terms after that term. Hence, we make sure that each passage has at least one term with a higher weight. The formula for defining the start and end of the passage is given below.

$$Start(Passage) = Position(t_k) - ((\frac{n}{2} - 1)\ if(AM(t_k) > threshold)$$

$$End(Passage) = Position(t_k) + ((\frac{n}{2})\quad if(AM(t_k) > threshold)$$

where, $AM(t_k)$ is the weight assigned to the term t_k, position(t_k) is the position of the term t_k in a particular document and threshold (0.4 for our dataset) is set empirically for each dataset.

4. EXPERIMENTAL SETUP

To validate our passage detection accuracy, we use the dataset used in [2], where each inserted passage within any document is tagged with a pre-defined category. The standard 20 Newsgroups (20NG) dataset is used that contains news articles about various topics such as sports, electronics, science, and more. Passages extracted from security related news articles on www.cnn.com are inserted into some documents (test documents) in the 20NG dataset. These documents are considered as "infected" documents. 18,000 documents from the 20NG dataset (20 categories) and 3,065 documents from this Security dataset (6 categories) [2] are used to train a text classifier. To ensure better performance of a text classifier, only the non-infected documents are used for training. In the testing phase, we use 1,000 infected documents and 1,000 non-infected documents.

To evaluate the effectiveness of our approach, we use the commonly used evaluation metrics of precision, recall and F1. Precision is defined as how accurately a system predicts whether a document contains a passage related to user specified category. Recall is defined as the ratio of number of correctly predicted documents that have hidden passages to the total number of documents that have hidden passages. F1 measure is a harmonic mean of recall and precision. We evaluate our approaches using two scenarios. In the first scenario, we consider true positive for an instance where a document is infected and is indeed detected as such. We call this task as *passage detection*. In the second scenario, we consider true positives for an instance only when the classifier correctly predicts the category of the passage in an infected document. We call this evaluation method as *passage category prediction*.

5. RESULTS

AM performed statistically significantly better than *odds ratio*, thus, we present only the results based on that. As depicted in Figure 1 and Figure 2, the *dynamic window* approach performs statistically significantly (99% confidence) better than methods presented in [2] with respect to F1 measure. As we only detect passages that contain terms with higher AM weight (i.e. less ambiguous terms), the number of false alarms significantly decreases and hence, the precision increases. As we define a new passage around each unambiguous term, the probability of detecting malicious passages increases. Thus, the recall value also increases. It was observed that using dynamic windowing significantly increases precision, recall and F1 measure for passage detection as well as passage category prediction as compared to other document splitting methods.

6. REFERENCES

[1] Kaszkiel, M., Zobel, J., Passage retrieval revisited. In Proceedings of the 20th annual international ACM-SIGIR conference on research and development in information retrieval (SIGIR 1997) Pg. 178-185.

[2] Mengle S., Goharian N, Detecting Hidden Passages from Documents. In proceedings of SIAM Conference on Data Mining (SDM 2008) Workshop, 2008

[3] Mengle S, Goharian N., Platt A., FACT: Fast Algorithm for Categorizing Text, In proceedings of IEEE 5th International Conference on Intelligence and Security Informatics, 2007, Pg. 308-315.

[4] Mengle S, Goharian N., Using Ambiguity Measure Feature Selection Algorithm for Support Vector Machine Classifier, In proceedings of ACM 23rd Symposium on Applied Computing, 2008, Pg. 916-920.