AN ITERATIVE UNSUPERVISED LEARNING METHOD FOR INFORMATION DISTILLATION

Kamand Kamangar[1] *Dilek Hakkani-Tür*[1] *Gokhan Tur*[2] *Michael Levit*[1]

[1]International Computer Science Institute (ICSI), Berkeley, CA 94704
[2]SRI International, Menlo Park, CA 94025
{kamand,dilek,levit}@icsi.berkeley.edu, gokhan@speech.sri.com

ABSTRACT

Information distillation techniques are used to analyze and interpret large volumes of speech and text archives in multiple languages and produce structured information of interest to the user. In this work, we propose an iterative unsupervised sentence extraction method to answer open-ended natural language queries about an event. The approach consists of finding the subset of sentences that are very likely to be relevant or irrelevant for the query from candidate documents, and iteratively training a classification model using these examples. Our results indicate that performance of the system may be improved by around 30% relative in terms of F-measure, by using the proposed method.

Index Terms— information distillation, unsupervised learning, question answering, machine learning

1. INTRODUCTION

The goal of an information distillation system is to extract an ordered set of segments called *snippets* that can be considered an answer to a given query from multi-lingual audio and text sources. A snippet can range from a fragment of a sentence to a paragraph. Below is an example query (in which the location and date range are variables) with some related snippets:

Query: *Describe attacks in [the Gaza Strip] giving location (as specific as possible), date, and number of dead and injured. Provide information since [28 Sept 2000].*

Snippets:

- *attack against a school bus filled with Israeli children*
- *There were 45 students and 2 teachers in the bus*
- *The militant Islamic Jihad claimed responsibility*

One critical component for distillation is detecting sentences to be extracted from each relevant document. The user typically is not interested in reading the whole news story, but instead just the sentences with the requested information content. The goal of sentence extraction is then to tag each sentence as relevant/irrelevant given a set of documents that are retrieved as relevant to a distillation query.

The queries are provided using predefined templates. In our previous work, we presented a data-driven method, named IXIR, for sentence extraction using lexical and name matching features for each template [1] and later extended that work to also include syntactic, semantic, information extraction (IE) annotation, and topicality features [2]. For example, the classification system for the example query type above may capture the patterns related to *attacks* from the previous related query and snippet pairs. Therefore, a training set is formed by marking snippets as relevant sentences in the corresponding documents, and all the rest of the sentences as irrelevant. A statistical classifier is then trained using this training data. When a new query is presented, first relevant documents to the query are retrieved by information retrieval (UMass INDRI search engine [3] in this case), and snippets are extracted using sentence extraction. A schematic representation of this algorithm is presented in Figure 1.

In this work, we propose an iterative unsupervised sentence extraction method to answer open-ended natural language queries about an event. We focus on answering the Template-1 queries of the DARPA GALE project [4], that look for responses to the query "Describe the facts about [EVENT]", with possible definitions of EVENT slot such as *"Looting of Iraqi Museums after U.S. invasion"*. Template-1 queries are harder to answer with our current distillation approach, which is based on supervised classification. Contrary to other templates, their form is very general, making it difficult to find trainable patterns for this template.

The IXIR system attempted to address this issue by using topicality features [2]. To obtain these features, we would compare sentences and slots in terms of the information they contain (each resulting in a separate topicality feature), such as word n-grams, IE-elements, semantic role labels etc. For instance, the value of a feature that is based on entity PER (person) is computed as the average instantiation score of all PER-entities found in the slot in the sentence. Co-reference, stemming and synonyms from WORDNET [5] and other means were employed to facilitate the instantiation.

The approach taken by the BBN Agile system depends on syntactic and semantic parsing [6]. The slot and sentences are

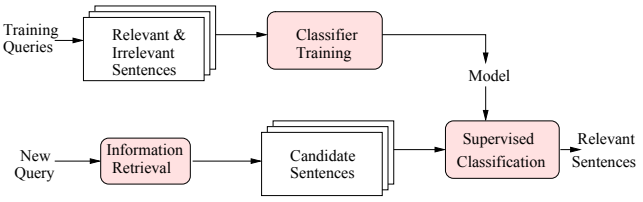

Fig. 1. Schematic representation of the baseline supervised distillation system

represented as a number of "proposition trees" and slot trees are instantiated in the sentence trees. Each proposition tree is a hierarchy of interconnected elementary predicate-argument structures ("propositions", [7]). Together they create what can be called a "meaning frame" of the text. Thus, instantiation of one proposition tree in another is tantamount to determining entailment of the slot in the sentence.

The proposed approach consists of building the sentence extraction models at runtime. It starts with training bootstrap models after finding the subset of sentences that are very likely to be relevant or irrelevant for the query, from the candidate documents. Then this process is iterated. In a way, this is similar to the self-training semi-supervised learning method. For example, for word sense disambiguation Yarowsky used a small *seed* set of examples to train an initial classifier, which is then used to label the remaining examples, and this classifier is iteratively improved [8]. Similarly McClosky *et al.* employed self-training for syntactic parsing and got 12% relative improvement [9]. Note that the approach we propose in this study is fully unsupervised and requires no labeled training data as the seed.

In the next section we present our approach describing the unsupervised machine learning for building models for GALE template-1 queries. In Section 3 we present experimental results using data for the DARPA-funded GALE program.

2. UNSUPERVISED LEARNING APPROACH

The idea of using feedback to improve search results goes back to [10]. In our paper we propose an on-the-fly unsupervised learning approach, where a small seed set of automatically labeled positive and negative examples is first extracted for initial classifier training. The training set is automatically expanded and refined in the course of several iterations. The performance of classifier highly depends on the accuracy of this feedback. While *blind feedback* [11] has been extensively studied for document retrieval, applying it to sentence extraction for information distillation via developing a statistical classifier is, to our knowledge, a novel approach.

Our main idea is as follows: Assume that we are given a collection of documents with the set of sentences S in these documents and a new query of the desired template. The goal is to find relevant sentences for that query. We start with a small set of sentences that have the highest likelihood of being an answer to this query. For example, these sentences may contain the exact wording of the query slot. We call this set A. Then we also find a set of sentences that have a very high likelihood of being irrelevant to the query, such as the ones that do not have any overlap in terms of words with the words in the query slot, after excluding the stop words. We call this set B. At this step we label the sentences of set A as relevant (1), and those of B as irrelevant (0).

We propose three different methods for the computation of the initial set of examples, that all use the stemmed non-stop words of the query slot:

1. Selection by total term frequency (TF): TF is the number of times a term appears in the candidate sentences. In this method, we first compute the total term frequency of all the stemmed non-stop words, and their mean frequency. Among these words, we select the ones that are more frequent than the mean frequency[1]. We then extract, from the original data, the sentences that contain *all* of these smaller subsets of words and label them as relevant, forming set A. Set B is then formed by extracting all the sentences that have *none* of the non-stop words and labeling them as irrelevant.

2. Selection by TF-IDF The term frequency multiplied by the inverse document frequency (IDF), commonly known as TF-IDF, is a weight often used in information retrieval [12]. IDF is usually computed as negative logarithm of the proportion of the documents containing the term. TF-IDF is a statistical measure used to evaluate the importance of a term for a document in a collection. The importance of a term increases proportionally to the number of times the term appears in the document but is limited by the frequency of the term in the document collection. In this method, we select the words with TF-IDF larger than a predefined threshold. This threshold is one of the parameters optimized according to a development set. As in the previous method, we then extract the sentences of the original data file that contain all of these smaller subsets of words and label them as relevant, forming set A. Set B is formed in the same way as in the previous method.

3. All terms considered equally important: In the previous methods, in some cases the classifier was not able to detect any relevant sentences for the initial step of classification. This happens when all the selected words are not present in individual sentences. This weakness prevents the learning of relevant sentences in the first iteration; consequently, the classifier never learns the correct features, even in the next runs. In other words, it leads to a zero value for recall and thus to a zero F-measure. Therefore, we used a method in which all words are considered as equally important. Contrary to previous methods where we started with a small collection of words, this time we start with the fewest words and make a more strict search in later runs. For example in the first itera-

[1] A more intuitive idea of considering less frequent words instead, fails due to the issue of data sparseness.

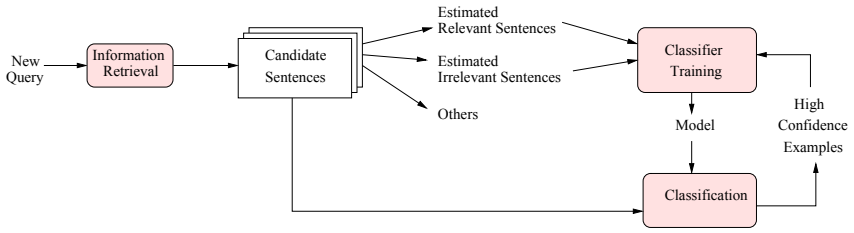

Fig. 2. Schematic representation of the proposed unsupervised distillation system

tion, we look for at least 10% of the query words to appear in a sentence and label such sentences as relevant. Then we label the sentences that do not carry any of the query slot words (after stop word removal and stemming) as irrelevant. We train an initial classifier, and proceed with the iterative step. Here, the percentage of words we look for is a parameter and is optimized according to the development set.

The next step is training a classifier iteratively with these high confidence examples, and then using this classifier to estimate the labels of examples in S or the examples in $S \setminus (A \cup B)$. The goal here, is to iteratively refine the classification decisions. After the initial step, the trained classifier is used in a self-training fashion to iteratively augment the training data. That is, we automatically classify the whole corpus. Each sentence in the corpus is weighted either to be relevant or to be irrelevant by a confidence output of the classifier. The next step is to select a set of confident sentences in both relevant and irrelevant classes. The confidence score threshold is another parameter that we optimize on the development set. We retrain the classifier using this new set. This makes the classifier learn words or phrases other than those already found. Then the same process is applied again. The number of iterations is also optimized on the development set. A schematic representation of this algorithm is presented in Figure 2.

For classification, we use the Boostexter classification tool [13], an implementation of the Boosting family of classifiers. But the approach is more general and classifier-independent.

3. EXPERIMENTS AND RESULTS

We performed n-fold cross-validation experiments using two sets of documents: the first set is formed by extracting the relevant documents from manually annotated answer keys provided by the LDC, and the second set is formed from the top 20 documents returned by the UMass INDRI IR engine as relevant to a given query. The IR engine output is then manually labeled in house by three annotators. We use 27 and 10 Template-1 queries from the GALE Y1 and Y2 data sets, respectively. The characteristics of these data sets are summarized in Table 1.

We use a development set to optimize the parameters of the algorithm, such as the number of iterations for which to run the algorithm. The performance of the proposed approach is computed using the F-measure. This is compared with two baselines: the original IXIR system performance and chance performance. The chance performance is obtained by selecting all sentences as relevant, hence resulting in 100% recall.

	LDC		INDRI	
	Y1	Y2	Y1	Y2
No. Queries	27	10	27	10
No. Documents	513	90	535	200
No. Irrel. Sents.	8,677	1,853	9,422	3,674
No. Rel. Sents.	2,295	494	2,079	977

Table 1. Properties of data sets used in the experiments: number of queries, documents, irrelevant sentences and relevant sentences in the GALE Y1 and Y2 data sets, when only relevant documents (LDC) and documents automatically retrieved are used (INDRI).

3.1. Supervised Classification

The performance of the supervised classification method is presented for the LDC documents in our previous work [2] for the 10 Y2 queries when the Y1 queries are used for training. The chance F-measure is 0.40 in this case. An F-measure of 0.42 is obtained when only words are used as features, and this performance is improved to 0.47 when an extended set of lexical, syntactic, semantic, and IE features is added.

3.2. Unsupervised Classification

We present our results using the proposed unsupervised learning methods for information distillation. Tables 2 and 3 summarize the results of n-fold cross-validation using 27 Y1 queries and the results when Y1 queries are used as development set with 10 Y2 queries as a test set, respectively. The two columns in both tables correspond to using only the relevant documents as extracted from LDC annotations (LDC), and the top 20 documents as returned by the information retrieval engine (INDRI). We obtain the best F-measure improvement using the all-terms method for both LDC and INDRI documents.

Method	F-measure (LDC)	F-measure (INDRI)
Chance	0.36	0.30
Term Frequency	0.38	0.32
TF-IDF	0.29	0.26
All Terms	**0.43**	**0.38**

Table 2. F-measure results after n-fold cross-validation on 27 Y1 queries, with all three methods using only relevant documents (LDC) and the documents returned by information retrieval (INDRI).

Method	F-measure (LDC)	F-measure (INDRI)
Chance	0.40	0.30
Term Frequency	0.36	0.25
TF-IDF	0.25	0.23
All Terms	**0.53**	**0.39**

Table 3. F-measure results on 10 Y2 queries, with all three methods using only relevant documents (LDC) and the documents returned by information retrieval (INDRI).

An analysis of the results shows that, when we use a selected subset of query terms, the number of relevant sentences at the first iteration is so small that it is hard to learn any meaningful patterns. Even though the *term frequency* method improves F-measure slightly with the LDC data, the improvement is lost when working with information retrieval output. The best method, *all terms*, considers all terms equally, and significantly improves performance on both data sets. The relative improvement for the cross-validation experiment is 17.6% for the LDC documents and 29.4% using the INDRI output. For the GALE Y2 queries, these numbers are 30.9% and 28.7% for LDC and INDRI documents, respectively. Performance of this method is also better than the F-measure reported in our previous work for the 10 GALE Y2 queries using the supervised training approach (0.53 versus 0.42 with lexical features, a relative improvement of 26%). Since the two methods are orthogonal to each other, combination of them is expected to result in further improvements.

4. CONCLUSIONS

We have presented an iterative unsupervised on-the-fly learning method for sentence extraction for information distillation. Our results indicate F-measure performance improvements of around 30% using the DARPA GALE queries. These results are also significantly better than our working system, where F-measure is 0.47.

In this work, we tested only English text documents of our corpus. However, the work can be extended to other languages, speech data, and other templates. Another important challenge would be to use the extended set of features as in the previous work rather than just words, such as information extraction annotations or syntactic or semantic features.

Acknowledgments: We thank Sebastien Cuendet and Dan Gillick for many helpful discussions. This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

5. REFERENCES

[1] D. Hakkani-Tür and Gokhan Tur, "Statistical sentence extraction for information distillation," in *Proceedings of the ICASSP*, Hawaii, April 2007.

[2] M. Levit, D. Hakkani-Tür, and G. Tur, "Integrating Several Annotation Layers for Statistical Information Distillation," in *Proceedings of the IEEE ASRU Workshop*, Kyoto, Japan, December 2007.

[3] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft, "Indri: A language-model based search engine for complex queries," in *Proceedings of the International Conference on Intelligent Analysis*, McLean, VA, May 2005.

[4] BAE, "Go/No-Go Formal Distillation Evaluation Plan for GALE," 2006.

[5] C. Fellbaum, *WordNet: An Electronic Lexical Database*, MIT Press, 1998.

[6] M. Levit, E. Boschee, and M. Freedman, "Selecting on-topic sentences from natural language corpora," in *Proceedings of the Interspeech*, Antwerp, Belgium, August 2007.

[7] R. Weischedel, J. Xu, and Licuanan A., "A Hybrid Approach to Answering Biographical Questions," in *New Directions in Question Answering*, M. Maybury, Ed., chapter 5, pp. 59–69. MIT Press, 2004.

[8] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proceedings of the ACL*, Cambridge, MA, 1995, pp. 189–196.

[9] D. McClosky, E. Charniak, and M. Johnson, "Effective self-training for parsing," in *Proceedings of the HLT/NAACL*, New York, NY, June 2006.

[10] J. Rocchio Jr., "Relevance feedback in information retrieval," in *The SMART Retrieval System: Experiments in Automatic Document Processing*, G. Salton, Ed., pp. 313–323. Prentice-Hall, Englewood Cliffs, NJ, 1971.

[11] G. Salton and C. Buckley, "Improving retrieval performance by relevance feedback," *Journal of the American Society for Information Science*, vol. 41, no. 4, pp. 288–297, 1990.

[12] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*, McGraw Hill, New York, 1983.

[13] R. E. Schapire and Y. Singer, "Boostexter: A boosting-based system for text categorization," *Machine Learning*, vol. 39, no. 2/3, pp. 135–168, 2000.