Exploiting Information Extraction Annotations for Document Retrieval in Distillation Tasks

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

[1]ICSI, Berkeley, CA
[2]SRI International, Menlo Park, CA
dilek@icsi.berkeley.edu, gokhan@speech.sri.com, levit@icsi.berkeley.edu

Abstract

Information distillation aims to extract relevant pieces of information related to a given query from massive, possibly multilingual, audio and textual document sources. In this paper, we present our approach for using information extraction annotations to augment document retrieval for distillation. We take advantage of the fact that some of the distillation queries can be associated with annotation elements introduced for the NIST Automatic Content Extraction (ACE) task. We experimentally show that using the ACE events to constrain the document set returned by an information retrieval engine significantly improves the precision at various recall rates for two different query templates.

Index Terms: information distillation, information retrieval, information extraction, document retrieval

1. Introduction

As the amount of available information grows tremendously, methods for directly accessing the relevant information efficiently and effectively become increasingly important. Now the need is for developing methods to extract only the requested information. In the framework of the DARPA GALE program, this process is called *distillation*. For example, given a set of multilingual audio and text sources, the purpose of distillation is to extract the biography of a person, or list arrests from a given organization during a specific time period with explanations. The participants are given a set of query templates with variable slots. The goal of a distillation system is to output ordered segments called *snippets* that can be considered as an answer to these queries. A snippet can range from a fragment of a sentence to a paragraph. Below is an example query from a template 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*

Our distillation approach is based on using document retrieval for finding relevant documents in response to a query, and statistical classification to extract sentences as snippets from the relevant documents [1]. The set of selected sentences is reduced by finding and eliminating redundancies. There are at least two possible ways of using information extraction (IE) annotations during this process. The first strategy employs IE annotations at the document retrieval stage where relevant documents are selected, while the second one uses these annotations to extract snippets (sentences) from selected documents. In this paper, we present our approach for using IE annotations to augment document retrieval by constraining the set of documents returned by information retrieval (IR) engine. By doing so, we take advantage of the fact that some of the distillation queries can be associated with annotation elements introduced for the NIST Automatic Content Extraction (ACE) task. For instance, two query templates that deal with arrest and attacks can be associated with the ACE events of the corresponding types. Therefore, ACE annotations can be used to narrow down the search for documents about these events.

In the experiments reported here, we incorporate IE annotations into this framework by intersecting the set of documents returned by the predominantly word-based IR engine with the set of documents that are consistent with the query in terms of their ACE annotations. We show that we can improve the document search results on two types of queries, namely GALE Information Distillation query templates 8 and 15, related to the ACE events, with the following general forms:

Query template 8: Describe the prosecution of [person] for [event].

Query template 15: Describe arrests of persons from [organization] and give their role in the organization.

In these query templates (QTs), the bracketed parts can have variable values as shown in the following examples:

Example (QT 8): *Describe the prosecution of Abu Abbas for the Achille Lauro Hijacking.*

Example (QT15): *Describe arrests of persons from PLO and give their role in the organization.*

In the next section, we describe related work on information distillation. We describe the ACE information extraction annotations in Section 3. Section 4 presents our distillation approach. In Section 5 we describe how we incorporate information extraction in document retrieval. Section 6 presents experiments and results using the TDT corpora.

2. Related Work

The distillation task is similar in nature to the question answering task that have been most extensively addressed by TREC evaluations [2]. Several participants used information extraction in their TREC systems. For instance, the system described in [3] used ACE relations among other *kernel facts* to find answers to definitional (“who/what is . . . ”) questions.

In TREC-6, Bear *et al.* presented a study in which they rerank the documents retrieved by an information retrieval engine (INQUERY) using an information extraction system (FAS-

TUS) [4]. More specifically, they checked if the named entities in the query appear in the documents returned. The scoring was done manually by giving a fixed score for each entity found.

Information extraction annotations have also been employed in the framework of GALE distillation. [5] used IE elements found in documents to reformulate the IR query. In the first pass, their system requests only high precision answers; then, ACE relations and events found in the returned documents are used to select relevant sentences; finally, words from these sentences are used to augment the original IR query.

Among other successful methods employed for question answering and distillation tasks are the logical proving of answer correctness using logical representations of a question and a putative answer as well as lexical axioms and world knowledge [6]. Finally, for questions relying on free-text topic formulations, the deep semantic representations of a question and a candidate answer that is based on the extracted predicate-argument structures can be embedded in an error-tolerant instantiation mechanism with the resulting instantiation score signalizing appropriateness of the answer [7].

The conjoint IE/IR strategy we present in this paper extends our existing system [1], where we use only IR to find documents related to a query before sentence extraction. Unlike [5], we use the IE annotations to constrain the set of documents returned by IR.

3. Information Extraction

The objective of the NIST ACE program is to develop automatic content extraction technology to support the automatic processing of source language data [8]. There are four primary ACE recognition tasks: recognition of *named entities, mentions, relations,* and *events*. Entities include person, location, organization, and other names. Mention extraction aims to group entities using nominal, pronominal, and named representations. Relation extraction tries to find predefined relations between the mentions such as *wife* of a person or *employee* of an organization. Event types include *Life, Movement, Transaction, Business, Conflict, Contact, Personnel,* and *Justice*. The *Justice* event type is closely related to query template 8. The *Justice* event subtypes include *Arrest/Jail*, which is closely related to query template 15. The *Conflict* event subtypes include *Attack*, which is closely related to query template 16.

We use the New York University (NYU) toolkit for information extraction [9], which provides entity and event annotations among others. The entities are extracted using a hidden Markov model based approach, in which each entity is represented with one state, and an extra state captures the non-entity tokens. The events are extracted using a combination of pattern matching and maximum entropy classification.

4. The Distillation Approach

All the data sources to be searched during the distillation task are determined in advance. The data includes both textual and audio data in multiple languages, namely, English, Chinese, and Arabic. We use automatic translations of the non-English data. For audio data we use both automatic and manual transcriptions. The University of Massachusetts INDRI search engine [10] indexes all the data. During runtime when a query is given, the INDRI search engine retrieves candidate documents, considering the dates, the sources of documents to be searched, and so on, as specified in the query. Then the sentence extraction process described in detail below tries to identify relevant sentences using the lexical and simple semantic information. Finally, similar sentences are clustered into groups. Because of the diversity of the data sources and the noise introduced via automatic speech recognition (ASR) and machine translation (MT), it is important to have a robust method.

4.1. INDRI Information Retrieval System

In this study we employ the INDRI information retrieval system [10] for document retrieval. INDRI employs an inference network approach, combining multiple evidences of relevance using statistical models. Therefore, the documents returned are associated with their relevance scores. Similarly, the arguments in the GALE distillation queries (such as the organization name in query template 15) are represented as nodes in the inference network and their appearances in the documents are scored in a similar fashion (with argument scores). If the query term, such as the name of a person, is missing in a document, then no argument score is returned. If only the first name of a person appears in the document, the score is also affected by the global frequency of that first name.

4.2. Sentence Extraction

Given a set of documents relevant to a distillation query as returned by document retrieval, the goal of sentence extraction is to tag each sentence in these documents with respect to its relevance. We employ a data-driven statistical method for sentence extraction in information distillation, and treat the problem as a binary classification task, where each sentence is classified as relevant (positive) or not relevant (negative).

To train the sentence extraction models, we extract negative and positive examples from the given answer keys, which have the relevant snippets and the corresponding document identifiers for each query. As the relevant sentences in those answer keys also include the document identifiers, we extract all sentences in those documents as examples, and mark the sentences whose portions are in the answer key as positive examples, and all the rest as negative examples. When answers are from non-English sources, we use the automatic translations of those answers as positive examples. This improves the robustness of the system to the noise introduced by ASR and MT [1]. For the experiments with ASR output we align the automatic hypotheses with reference sentences and extract their class (positive or negative) from the answer keys.

During classification we use lexical and simple semantic features. Lexical features consist of word n-grams obtained from the training examples. This can be considered as a query-specific information extraction system, which is supposed to perform better than a generic one. We then augment these features with semantic ones by tagging the raw sentences to mark instances of the organizations, locations, or dates in the query. Sometimes equivalent terms are also given in the query. For example, the equivalent term for the organization *al-Qaeda* is *al-Qaida*. Similar mapping is done for those phrases as well. Then the classification features include the word and/or tag n-grams extracted from both the raw sentence and the tagged sentence.

We use the Boostexter classification tool [11], an implementation of the Boosting family of classifiers, though note that this approach is independent of the specific classification algorithm used.

5. Using Information Extraction Annotations in Document Retrieval

As mentioned in the previous section, like most other information retrieval systems, INDRI is task independent. On the other hand, sentence extraction, as the name implies, uses sentences as the basic units. When a distillation query is submitted to this IR engine, it is up to the distillation engine to determine the number of documents that need to be returned. This is problematic since it is hard to know the optimal value that holds for all queries. Sometimes a query has only one relevant document in the huge document repository and sometimes thousands. If the sentence extraction system processes a larger number of returned documents, this results in a higher number of false alarms unless document level processing is available. One solution might be getting fewer documents from INDRI but this may result in poorer recall. Alternatively one could exploit the document and argument scores returned by INDRI. However the document and argument scores have different dynamic ranges depending on the query and it is not easy to perform thresholding that works optimally for all queries using them. One may also filter out the documents for which no argument score is returned, indicating that the argument does not appear in the document. However this alone does not indicate that the document is irrelevant. For example if the query is asking about the events in Iraq and the document is about a bombing in Baghdad, it is relevant. Similarly finding the argument mentioned in the document does not indicate that the document is relevant.

For these reasons we propose having an intermediate processing stage between the INDRI information retrieval engine and the sentence extraction module, to filter out irrelevant documents. The basic idea is as follows: Since the distillation query templates are known beforehand, it is sometimes possible to associate expected document contents with one or several types of ACE annotations. For example, for the query template:

> Describe attacks in [location] giving specific location, date, and number of dead and injured between [dates] .

the relevant document must have the ACE event of subtype *attack* and the location mentioned in the query. Since the information extraction system provides them both, the postprocessing stage needs to check only whether the locations mentioned in the query (or their equivalences) also appear in each document. Figure 1 depicts the basic idea of exploiting information extraction to improve document retrieval.

6. Experiments and Results

To check the usability of IE annotations, we conducted document search experiments using the INDRI toolkit with and without the IE annotations for 34 queries for two GALE distillation query templates: 8 (about prosecutions of people) and 15 (about arrests of members of an organization). There are 13 queries for template number 8, and 21 queries for template number 15. The answer keys, formed from a set of documents that are considered for these queries, and snippets corresponding to relevant documents are provided by GALE (reference annotations are produced by human labelers at LDC and BAE systems). These documents are selected from the LDC TDT-4 and TDT-5 corpora. However, the annotations include relevance judgments of labelers for only a subset of all the documents. The average number of relevant documents annotated for template number 8 is 21.4, and for template number 15 is 24.3.

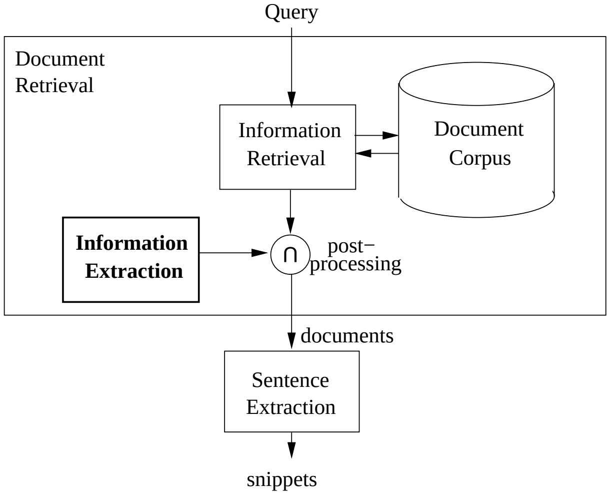

Figure 1: Use of IE to improve document search for information distillation.

Best F-measure	**template 8**	**template 15**
IR (fixed M docs)	52.8%	38.7%
IR (doc. score threshold)	52.2%	43.8%
IR (arg. score threshold)	50.5%	43.8%
IR+IE	58.3%	46.7%

Table 1: Best F-measure values for different methods.

To avoid dealing with ASR/MT noise in these experiments, we limited the document space to English documents from TDT-4 and TDT-5 newswire sources. To evaluate our approach, we extracted the top N ($N = 50$) documents returned by the INDRI toolkit, and manually annotated the documents that were not already marked in the answer keys provided. Out of 1700 documents returned by the INDRI engine, for 34 queries, 684 were already manually labeled by BAE with relevance judgments (306 relevant, 378 not relevant).

We compute micro-averaged recall and precision curves by selecting the best M out of N documents returned by INDRI ($M = 1, 10, 20, 30, 40, 50, N = 50$ as only the top 50 documents are manually annotated). Since the relevance of each document from TDT-4 and TDT-5 can not be easily manually annotated with respect to each query, the precision numbers quoted in the recall and precision plots are accurate, but the recall numbers are scaled up. As the total number of relevant documents for each query is the same for the two methods, the relative improvements in recall at a given precision rate hold, despite the fact that there might be documents that are relevant to a query other than the ones in answer keys provided by BAE or the top N documents returned by INDRI.

First we evaluated three methods of selecting documents from the IR output: *fixed number of documents, document score thresholding,* and *argument score thresholding*. In the first case, the top M documents are used for each query. In document score thresholding, the documents that have a score higher than a threshold T_1 are selected. Argument score thresholding is similar to document score thresholding; however the argument score is used instead of the overall document score. To check the effect of using IE annotations, we chose the first method of document selection and constrained the set of selected documents to contain the annotation of the event corresponding

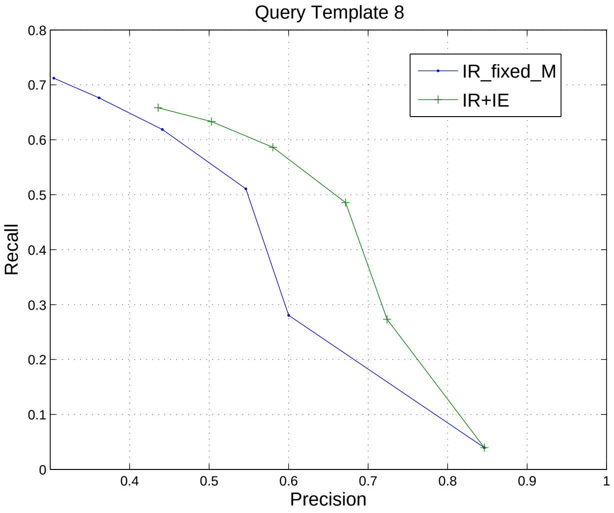

Figure 2: *Recall and precision for query template 8.*

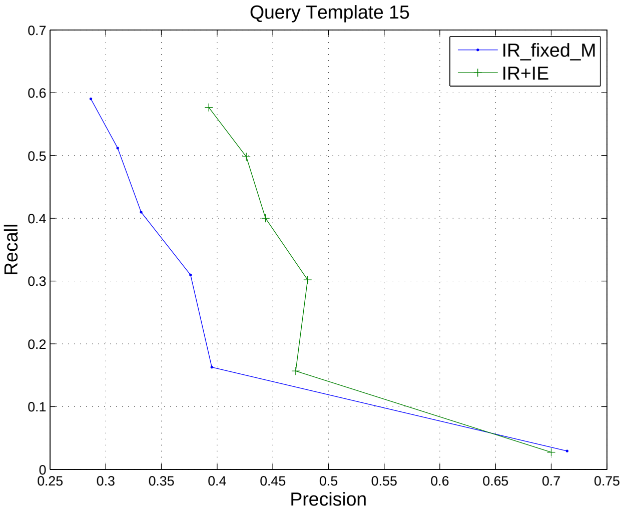

Figure 3: *Recall and precision for query template 15.*

to the query template in question. The best F-measure numbers (that have been obtained by optimizing parameters) with all methods are shown in Table 1 for all four methods. As shown, the method using the IE annotations results in the best F-measure. For query templates 8 and 15 the performance significantly improved by 10% relative and 20% relative. Document and argument score thresholding resulted in mixed results for different query templates. This mayoccur because scores for person names are suboptimal for this task. For example, finding the first name of a person is a weak indicator of relevance.

We also plot recall and precision curves for the fixed number of IR documents method, and the combination of IE and IR approach for query templates 8 and 15, in Figures 2 and 3, respectively. As shown in these plots, we obtain significant improvements in recall at various precision rates for two different query templates, using the IE aided approach.

7. Conclusions

We presented our approach for using information extraction annotations to augment document retrieval, where we took advantage of the fact that some of the distillation queries can be associated with annotation elements introduced for the NIST ACE task. We have experimentally shown that, when we use ACE events to constrain the document set returned by IR, we obtain significant improvements in precision at various recall rates for two different query templates related to ACE “arrest” and “justice” events. As future work, we plan to incorporate other ACE annotations to improve document retrieval, and also use these annotations to expand the feature set for sentence extraction.

8. Acknowledgments

We thank SRI GALE Distillation team members at Columbia University, University of Massachusetts, NYU, and Fair Isaac. 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.

9. References

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

[2] E. M. Voorhees, “Overview of the TREC 2003 question answering track,” in *Proceedings of the Twelfth Text REtrieval Conference (TREC-2003)*, 2003.

[3] J. Xu, “TREC 2003 QA at BBN: Answering definitional questions,” in *Proceedings of the Twelfth Text REtrieval Conference (TREC-2003)*, 2003.

[4] J. Bear, D. Israel, J. Petit, and D. Martin, “Using information extraction to improve document retrieval,” in *Proceedings of the Sixth Text Retrieval Conference (TREC-6)*, 1997.

[5] B. Schiffman, K. McKeown, R. Grishman, and J. Allan, “Question answering using integrated information retrieval and information extraction,” in *Proceedings of HLT/NAACL*, Rochester, April 2007.

[6] D. I. Moldovan, C. Clark, S. M. Harabagiu, and S. J. Maiorano, “Cogex: A logic prover for question answering,” in *Proceedings of the HLT-NAACL*, 2003.

[7] M. Levit, E. Boschee, and M. Freedman, “Selecting ontopic sentences from natural language corpora,” in *Proceedings of the Interspeech 2007*, Antwerp, Belgium, 2007.

[8] NIST, “The ACE 2007 (ACE07) evaluation plan,” NIST, Tech. Rep., 2007, http://www.nist.gov/speech/tests/ace/ace07/doc/ace07-evalplan.v1.3a.pdf.

[9] R. Grishman, D. Westbrook, and A. Meyers, “NYU’s English ACE 2005 system description,” NYU, Tech. Rep. 05-019, 2005.

[10] 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.

[11] 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.