A Hybrid Approach for QA Track Definitional Questions

Sasha Blair-Goldensohn **Kathleen R. McKeown** **Andrew Hazen Schlaikjer**
Department of Computer Science
Columbia University
New York, NY 10027
{sashabg,kathy,hazen}@cs.columbia.edu

Abstract

We present an overview of DefScriber, a system developed at Columbia University that combines knowledge-based and statistical methods to answer definitional questions of the form, "What is X?" We discuss how DefScriber was applied to the definition questions in the TREC 2003 QA track main task. We conclude with an analysis of our system's results on the definition questions.[1]

1 Introduction

In recent years, the QA systems in TREC have reached a remarkably high level of performance (Voorhees, 2002). Until this year, however, the task has focused on the short-answer, or *factoid* model, in which the goal is to answer questions for which the correct response is a number, short phrase, or sentence fragment. In this paper, we focus on the newly introduced *definitional* question type and present the results of a system which we have built to answer such questions.

Why build a system that is specific to definitional questions? Consider a student asked to prepare a report on the Hajj, an Islamic religious duty. In the context of short-answer QA, both patience and prescience will be required to elicit the core facts. First, a relatively long list of sub-questions would be required (e.g., "Where is the Hajj carried out?", "How long does it last?", "Who undertakes a Hajj?" etc.). Second, knowing which questions to ask requires knowledge that the questioner likely does not have. That is, the questions that best elicit a description of one thing (e.g., the Hajj) can be quite different from those best suited for finding out about something else (e.g., the Caspian Sea).

Instead, it is useful to have a sytem which can answer "What is X?" questions directly, presenting a comprehensive response which effectively combines the answers to the relevant sub-questions. This capability is a valuable complement to static knowledge sources like encyclopedias, especially in answering questions about an "X" whose meaning may be evolving, or in creating custom answers that focus on particular aspects of a definition.

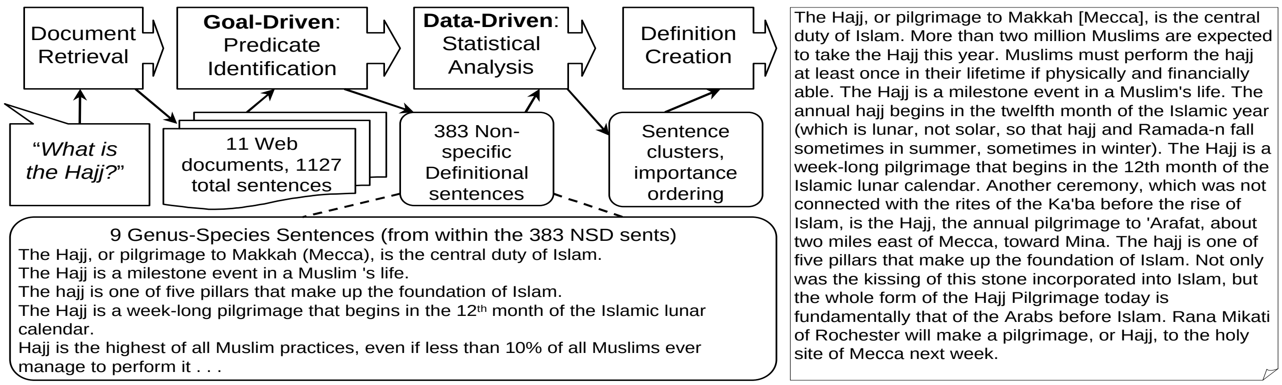
The remainder of this paper presents DefScriber, a definitional QA system implemented at Columbia University. We first present a brief overview of the system's architecture and previous evaluations (more detail on these topics has been reported previously (Blair-Goldensohn et al., 2003)). We then focus on our performance on the 50 definitional questions included in this year's TREC QA main task.

2 Architecture Overview

Figure 1 gives a high-level view of DefScriber's operation, illustrating input and output of each stage. This example traces an actual answer generated for the question "What is the Hajj?"

The input is specified as a question, which feeds into the document retrieval phase. The user may specify which databases to search, a maximum number of documents to retrieve, and the desired answer length.

[1] We did not produce answers to the other, non-definitional, questions.


Document Retrieval
Goal-Driven: Predicate Identification
Data-Driven: Statistical Analysis
Definition Creation
"*What is the Hajj?*"
11 Web documents, 1127 total sentences
383 Non-specific Definitional sentences
Sentence clusters, importance ordering
9 Genus-Species Sentences (from within the 383 NSD sents)

The Hajj, or pilgrimage to Makkah (Mecca), is the central duty of Islam.
The Hajj is a milestone event in a Muslim 's life.
The hajj is one of five pillars that make up the foundation of Islam.
The Hajj is a week-long pilgrimage that begins in the 12th month of the Islamic lunar calendar.
Hajj is the highest of all Muslim practices, even if less than 10% of all Muslims ever manage to perform it . . .

The Hajj, or pilgrimage to Makkah [Mecca], is the central duty of Islam. More than two million Muslims are expected to take the Hajj this year. Muslims must perform the hajj at least once in their lifetime if physically and financially able. The Hajj is a milestone event in a Muslim's life. The annual hajj begins in the twelfth month of the Islamic year (which is lunar, not solar, so that hajj and Ramada-n fall sometimes in summer, sometimes in winter). The Hajj is a week-long pilgrimage that begins in the 12th month of the Islamic lunar calendar. Another ceremony, which was not connected with the rites of the Ka'ba before the rise of Islam, is the Hajj, the annual pilgrimage to 'Arafat, about two miles east of Mecca, toward Mina. The hajj is one of five pillars that make up the foundation of Islam. Not only was the kissing of this stone incorporated into Islam, but the whole form of the Hajj Pilgrimage today is fundamentally that of the Arabs before Islam. Rana Mikati of Rochester will make a pilgrimage, or Hajj, to the holy site of Mecca next week.

Figure 1: DefScriber answers "What is the Hajj?"

Currently, DefScriber is able to search the Internet via Google, as well as the TREC-11 and CNS[2] collections, which are indexed locally.

The information retrieval (IR) module uses a fixed set of patterns to identify the term to be defined in the question, and then generates a set of search queries. These queries are sent to the selected search engine in order of decreasing expected precision until a threshold number of documents has been retrieved or the set of queries has been exhausted.

Once documents are retrieved, the primary goal-driven step is performed, with the system examining documents for instances of definitional predicates. Next, a data-driven analysis produces sentence clustering and ordering information. In the last step, a definitional answer is created via sentence extraction, guided by the results of the goal- and data-driven stages.

3 Definitional Predicates: A Goal-Driven Approach

Answering a "What is X?" definitional question and creating a summary of query results for the search term "X" are strongly related problems. Yet, as readers, we have more specific expectations for a definition than for a general-use summary. The idea of definitional predicates is to model these special properties of a definition so the system can use them to create better answers.

3.1 The Predicate Set

Our set of definitional predicates is shown in Table 1. Currently, the system automatically identifies instances of three of these types in text: Genus, Species and Non-specific Definitional (NSD). Research on identifying Target Partition and History instances is ongoing.

An important distinction is that NSD subsumes all of the other more specific predicate types that appear underneath it in Table 1. Thus, identifying NSD text is crucial because it is a cue to the presence of other predicates; it also removes noise and provides a set of useful definitional text which is given as input to data-driven methods even when the text cannot be further classified with a more specific predicate. We chose Genus and Species as the first specific predicates to implement because they are at the core of what definitions are; Related work (Sager and L'Homme, 1994; Swartz, 1997; Sarner and Carberry, 1988) consistently identifies these two concepts as key parts of defining a term.

3.2 Automatic Predicate Identification

To use these predicates in our system, we must identify units of text which contain them. To do this, we first did a manual examination of documents to create sample data annotated with predicates. Using this data, we explored two approaches to identifying predicates. The first uses machine learning to learn a feature-based classifier that predicts when a predicate occurs. The second uses pattern-recognition over patterns extracted from the annotated data.

We used the machine learning approach to automatically identify NSD sentences. Using a set of

[2] A collection of documents from the Center for Nonproliferation Studies (`http://cns.miis.edu`) made available to participants in the AQUAINT project.

Predicate	*Description*	*Instance Example*
Non-specific Definitional	Any type of information relevant in a detailed definition of the term. NSD are a superset of the below predicates.	Costs: Pilgrims pay substantial tariffs to the occupiers of Makkah and the rulers of...
Genus	Category to which term belongs.	The hajj is a type of ritual.
Species	Describes properties other than or in addition to Genus. Species are a superset of the below predicates.	The annual hajj begins in the twelfth month of the Islamic year.
Target Partition	Divides the term into two or more conceptual or physical parts.	Qiran, Tammatu' and Ifrad are three different types of Hajj.
Cause (effect)	States explicitly that the term is the cause (effect) of something.	The pilgrimage causes the past sins of a Muslim to be forgiven.
History	Gives historical information relating to the term.	Mohammed, founder of Islam, started the tradition in 632 C.E.
Etymology	Information on the term's genesis, e.g. adaptation from another language.	In Arabic, the word Hajj means a resolve of magnificent duty.

Table 1: Definitional Predicates: Descriptions and Examples

surface features such as sentence position (relative and absolute in a document) "term concentration" (i.e. the term's frequency within a sentence and/or nearby sentences), we applied two machine learning tools: the rule-learning tool Ripper (Cohen, 1995) and the boosting-based categorization system BoosTexter (Schapire and Singer, 2000). Both algorithms performed similarly in terms of the accuracy of their predictions on test data; Ripper's rules are used in DefScriber since they were somewhat simpler to implement. Using cross-validation, accuracy of 81 percent was obtained with Ripper (76 percent using BoosTexter).

In order to identify Genus and Species predicates, we manually extracted a set of lexicosyntactic patterns to model sentences containing both Genus and Species (G-S) information, as these G-S sentences provide a strong grounding context for understanding the term. Rather than modeling the patterns at the word level, i.e. as flat templates with slots to fill, we model them as partially specified syntax trees (Figure 2). One such pattern can match a large class of syntactically similar sentences without having to model every type of possible lexical variation. This approach derives from techniques used in information extraction (Grishman, 1997), where partial subtrees for matching domain-specific concepts and named entities are used because automatic derivation of full parse trees is not always reliable. However, data-driven techniques (Section 5) offer additional protection from false or extraneous matches by lowering the importance ranking of information not corroborated elsewhere in the data.

Figure 2 illustrates the transformation from example sentence to pattern, and then shows a matching sentence. Our patterns are flexible - note that the example and matched sentences have somewhat different trees. Another point of flexibility is the verb itself; FormativeVb will match verbs in a set which our algorithm considers expressive of "belonging" to a category (e.g., "be," "represent," "exemplify").

Using our predicate-annotated data set, we have manually extracted 23 distinct patterns which match G-S sentences. Although it is difficult to reliably measure recall of the patterns without a larger set of annotated documents, precision in previous evaluations was approximately 96 perecent.

4 Data-Driven Techniques: Applying Summarization

While our set of predicates, including Genus and Species, are domain-neutral, they are not meant to model all possible important information for a given term definition. Some information types may be hard to define computationally *a priori*. Also, a given sentence may instantiate a definitional predicate but include only peripheral content. We address these issues in the data-driven stage of DefScriber's pipeline (Figure 1), applying statistical techniques adapted from multi-document summarization to the Non-specific Definitional sentences identified in the goal-driven stage.

First, a definition centroid is computed by creating a stemmed-word vector of all the NSD sentences. Then the individual sentences are sorted in order of decreasing "centrality," as approximated by IDF-weighted cosine distance from the definition centroid. This method creates a definition of length N by taking the

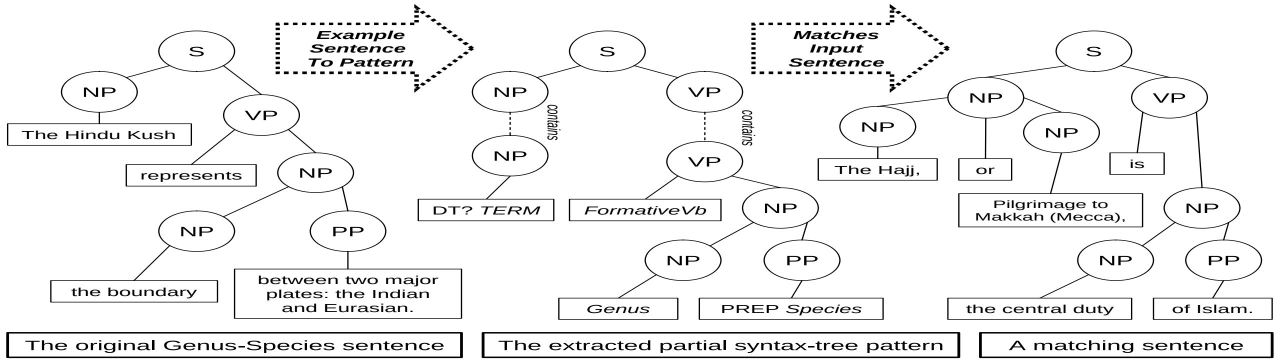


Figure 2: Pattern extraction and matching for a Genus-Species sentence from an example sentence.

first N unique sentences out of this sorted order, and serves as the TopN baseline method in our evaluation. Note that this method approximates centroid-based summarization, a competitive summarization technique (Radev et al., 2000).

After ordering sentences with TopN, we perform a non-hierarchical clustering that we use to decrease redundancy by avoiding same-cluster sentences in the answer. Since our clustering similarity measure uses IDF computed over a large collection, it can suffer from overweighting of specialized terms; to account for this, we augment the cosine distance calculation, using local IDF values calculated dynamically from the pool of NSD sentences.

The final data-driven technique improves cohesion by considering the content of the previous answer sentence when choosing a sentence to add to the answer. After choosing the first sentence as in TopN, we choose each remaining sentence as follows: we define the goodness of a cluster as an equal weighted combination of (1) its cohesion to the previous sentence and (2) its overall importance, approximated by that cluster's centroid's distance from (1) the previously chosen sentence's cluster's centroid and (2) the centroid of all NSD sentences. We also add in a penalty for clusters from which sentences have already been chosen such that no cluster gets n sentences included in the answer before all clusters have $n-1$ included sentence. Once the "best" next cluster has been chosen in this manner, we add the next sentence to the definition as the top-ranked sentence in that cluster which has not yet been included in the definition.

DefScriber's default configuration integrates all the above data-driven techniques (TopN, clustering, local IDF weighting, and cohesion ordering), combining them with the goal-driven method of G-S predicate identification. We place the top-ranking (in terms of TopN) G-S sentence first in the definition, and use the cohesion-based ordering to add the remaining sentences. We call this integrated goal- and data-driven method DefScriber.

5 Related Work

Goal-driven, or top-down, approaches are more often found in generation. Schemas (McKeown, 1985), rhetorical structure theory (Mann and Thompson, 1988; Moore and Paris, 1992; Hovy, 1993; Marcu, 1997) and plan-based approaches (Reiter and Dale, 2000) are examples of goal-driven approaches, where the schema or plan specifies the kind of information to include in a generated text. In early work, schemas were used to generate definitions (McKeown, 1985), but the information for the definitional text was found in a knowledge base. In more recent work, information extraction is used to create a top-down approach to summarization (Radev and McKeown, 1998) by searching for specific types of information which can be extracted from the input texts (e.g., perpetrator in a news article on terrorism). Here, the summary briefs the user on domain-specific information assumed a priori to be of interest.

Other long-answer QA systems are currently under development as part of the AQUAINT program (Voorhees, 2003). Some of these share attributes with DefScriber; Weischedel et al.(ARD, 2003) explore definitional and biographical questions, using a combination of methods that are largely complementary to those used in DefScriber, namely identification of key linguistic constructions and information extraction (IE) to identify specific types of semantic data.

Another important contrast between DefScriber and most of the long-answer systems developed under the AQUAINT program has to do with answer format. While these systems mostly produce answers as a ranked list of descriptive phrases or sentences, DefScriber uses summarization methods to produce a coherent, multi-sentence, encyclopedia-style definition.

6 Previous Evaluations

An evaluation of DefScriber performed previously used human judgments to measure the performance of DefScriber's definitions over a set of 24 terms from a varied set of domains. We measured five qualities of the definitions: relevance (precision), redundancy, structure, breadth of coverage, and term understanding. Overall, we found that DefScriber achieved the best scores in structure, redundancy, term understanding, and relevance, with statistically significant margins in the first two categories. In coverage, DefScriber performed below the baselines, but not at a statistically significant level. The results are reported in detail elsewhere (Blair-Goldensohn et al., 2003).

7 Modifications for TREC 2003

Although DefScriber is built specifically to answer definitional questions, several modifications and optimizations were performed before running the system for the definitional questions in the TREC QA task.

First, the data source to use needed to be fixed: usually, a user of DefScriber's web interface would specify whether to query Internet documents or local collections. For the TREC question set, we hard-coded this value so that only the TREC-11 data sets were searched.

Secondly, we needed to reconsider our metric for a "good" answer in light of the announced scoring formula. Since each answer for a definition question was to be considered on the basis of its intrinsic information content alone, the statistical cohesion measures described in Section 4 were disabled. Clustering was still used to avoid redundancy, since redundant information nuggets would receive a zero score. But answer sentences were picked from the clusters in a purely importance-based order (as approximated by our TopN ordering), without regard to cohesion.

Another point of modification was the issue of handling "Who is X?" as opposed to "What is X?" questions, since both types were included in the TREC definition question set. Although DefScriber has been designed primarily to provide definitions of objects and concepts[3], its design allows "Who" questions to be processed easily as well. In fact, a look at the predicates in Table 1 reveals that they can be applied to people, for instance we can and do identify the sentence, "John Glenn was the first astronaut." as a G-S sentence even though its subject is a person. The single difference in DefScriber's processing of "Who" as opposed to "What" questions is that sentences which include certain personal pronouns like "he" or "she" do not have their score reduced as they would for a "What" question.

Lastly, we needed to decide how many answer sentences to include for each definitional answer. Our current system takes this number as a user-specified parameter, but in this case we needed the system to try to determine an optimal value. Our approach was to use the training data provided by the AQUAINT pilot study (Voorhees, 2003), and to optimize a linear combination of a base answer length and an adjustment factor based on the number of relevant documents (i.e. containing one or more NSD sentence) found for a particular answer. We did this by using the assessor nuggets for the 25 pilot definition questions, and calculating what our score would have been if our answer length were determined as a linear function of the number of relevant documents found. We approximated this optimum as:

$$\underset{base, factor \in 1..20}{max} \; \underset{q \in 1..25}{avg} \; (F(q, base + docs(q)/factor))$$

Where $F(q, n)$ is the TREC F-measure score for DefScriber's n-sentence answer on pilot question q, and $docs(q)$ is the number of relevant documents found for question q. The optimum was found at $base = 9, factor = 16$. Therefore, the final modification of our system for the TREC task was to set it to produce $(9 + docs(q)/16)$-sentence answers for each definitional question q.

8 Performance on TREC 2003 Definition Questions

As mentioned previously, our system is designed specifically to answer definitional questions and as

[3] This is in part because a separate, complementary system with greater focus on properties specific to describing individuals, i.e. biographies, is under development at Columbia (Duboue and McKeown, 2003)

QID	*Question*	*Official Nugget*	*Matching Response?*
1901	Who is Aaron Copland?	established home for composers	Music from the Copland House made its debut Sept. 29 at Merkin Concert Hall with, appropriately, an all-Copland program.
1905	What is a golden parachute?	Agreement between companies and top executives	William M. Mercer Inc., the consulting firm, has found that 64 percent of 350 large publicly traded companies provide financial protection for one or more key executives, most often the chief executive, if the company changes control.
2274	Who is Alice Rivlin?	vital financial assistance Authority for DC	She was too busy overseeing the city government of the District of Columbia as chairman of its financial control board – and serving as vice chairman of the Federal Reserve Board in her day job – to accept the plaque.
1907	Who is Alberto Tomba?	two time world cahmpion	Italian Alberto Tomba, three-time Olympic and two-time world champion, came back to win the last World Cup slalom in Schladming, Austria, on Thursday before the world Alpine championships.

Table 2: Sentences from DefScriber's output that were judged as non-matching alongside possibly matching nuggets. These non-matches demonstrate the ambiguous nature the judgement matching proces; the first two sentences are arguable matches but require significant inference on the judge's part; the latter two seem to be clearer matches.

such was run only for the definition questions in the main QA track. Thus we will focus on our results for these questions.

Overall, our system performed well above the median for these questions, achieving an average F score of .338 compared with the median of .192. Examining the evaluation results further, we made a number of observations about the evaluation in general and our performance in particular.

An initial study of evaluation results showed that some data nuggets present in our response sets were not counted by judges, resulting in degraded recall scores. These judgements may be due in part to the need for higher level inference over response sentences and nuggets to see their connections. The issue of whether such inference is appropriate may have been a source of considerable noise in the evaluation. Table 2 gives several examples of response sentences from our output which were not scored as containing an official nugget, alongside the nugget which they might arguably have been matched with, indicating potential judgment errors. These examples are meant to show the gradient of answer-matching ambiguity from more to less ambiguous. That is, the first two answers would seem to require some level of inference on the part of the judge to be certified a match; the latter two seem more clearly to include the desired nugget.

A strongly related issue, particularly for a system like DefScriber which produces a multi-sentence answer meant to be read as a whole, is the issue of many-to-one matches. That is, should judges count a nugget as "matching" when its information is not contained in a single answer sentence, but rather in the sum of information provided by several answer sentences?

DefScriber also encountered difficulty with certain questions because of its reliance (at the time of the evaluation) the MG search engine, which lacks support for phrasal queries. Lack of phrasal search resulted in low precision, coupled with limitations on the number of documents processed resulted in low recall. This problem became more pronounced in cases where one or more words in term/person to be defined was common, resulting in a large set of documents being returned from MG, which does a boolean OR across all query words. Due to speed limitations, our system truncated such large results sets at a fixed size, and thus found only a subset of the documents which actually contained the term words in a phrase. This resulted in problems, for instance, on questions of the type "Who is X?", where X had a very common first and/or last name (e.g. "Al Sharpton", and "Andrea Bocelli"). Subsequently, updates have been made to DefScriber's IR module so that it now fully supports phrasal search capabilities on locally indexed corpora via the Lucene search engine.

For questions where this was not an issue, DefScriber's sentence selection criteria seem to have performed well, both goal- and data-driven. In some of our higher scoring answers, we see an impact from our goal-driven strategy via the identification of G-S sentences; for instance, we can see in Figure 3 that Def-

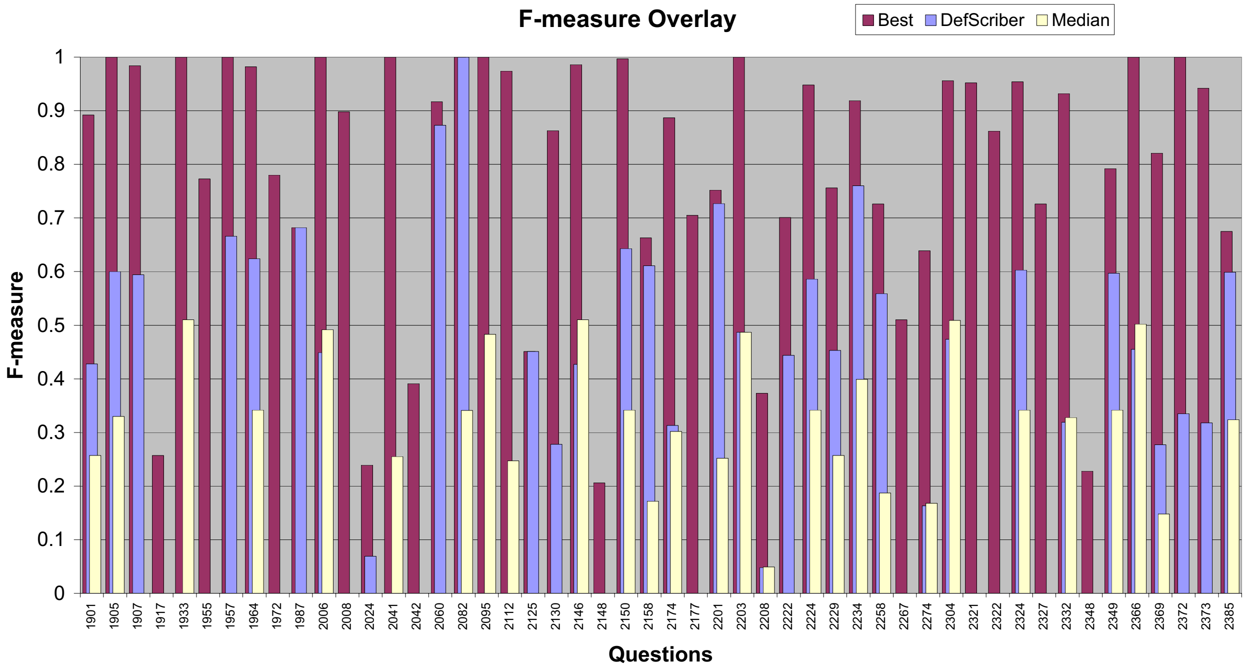


Figure 3: Overlay of evaluation-wide F-measure scores per question with those of DefScriber

Scriber achieves the highest score on question 1987, “What is ETA in Spain?”, which appears to be one of the harder questions in that its median score was zero. Our high score on this question is in part due to finding and including the G-S sentence, “ETA is an acronym for Basque Homeland and Freedom in the Basque language.”, which contains one of the “vital”information nuggets for this question, i.e. the information on what ETA stands for.

While G-S sentences are clearly helpful, we were also successful when G-S sentences were not found. In these cases, we rely on our robust data-driven methods to statistically guide answer content. These methods allowed us to select high-scoring sentences even when G-S sentences were not found. Such instances included our best and near-best scores on questions 2060 (“Who is Alberto Ghiorso?”), 2082 (“Who was Anthony Blunt?”), 2125 (“Who was Charles Lindbergh?”) and 2201 (“What is Bollywood?”).

However, even when IR returned relevant documents, we did see degradation in system performance where high numbers of response sentences were returned. We believe this is due to the precision penalties our system suffered by using the adjustable answer-length threshold explained in Section 7. Since this threshold creates a longer answer when more relevant sentences are found, our lower scores in these cases suggest that the penalties we incurred in precision did not make up for whatever additional recall nuggets we achieved by having longer answers; it would be interesting to see if the answer-length optimization described in the Section 7 would arrive at a smaller length function given the new data from this evaluation.

As suggested by the zero median F-measure of question 2024 “Who is Andrea Bocceli”, few participants in the evaluation have incorporated fuzzy search capabilities to overcome spelling errors in input questions (the singer’s name is correctly spelled “Bocelli”). From an IR prespective, this represents a very important advance that most systems should make in order to function adequately with noisy data from source materials and/or search inputs.

For future evaluations where nuggets of information are to be identified by human judges, it may be useful to perform some error analysis of adjudications made this year. Given the subjective nature of the task, attaining a “perfect” scoring is of course impossible. But an analysis of the kinds of errors or issues seen will be important as we seek to refine the design of the definition question task and the judgement process itself.

9 Future Work

Future work on DefScriber will concentrate on increasing the number of definitional predicates automatically identified by the system, as well as on improving identification performance on such predicates.

We are currently working to improve our feature-based predicate identification methods by growing our annotated data set while also extracting more and richer features to input into our machine learning methods. To improve the pattern-based methods, we are actively working with IE bootstrapping techniques developed in Snowball (Agichtein and Gravano, 2000) to automatically learn predicate patterns from manually extracted "seed" examples. Such techniques would allow us to supplement our manually-generated patterns and bring new predicates online more quickly.

10 Conclusion

We have presented an overview of DefScriber, a hybrid goal-driven and data-driven system for definitional questions. We explained how the system was modified and applied to answer definitional questions in the TREC 2003 QA track. Finally, we presented DefScriber's results on the definitional questions, which were significantly above median, achieving an average F-score of .338 compared with the median of .192. Finally, we analyzed our scores on certain individual questions, discussing areas where our system performed well and others where it could be improved, as well as noting several issues of the judgement process itself.

Acknowledgements

The authors wish to acknowledge the support of ARDA through AQUAINT contract MDA908-02-C-0008. We are also thankful for the generous and thoughtful contributions of colleagues at Columbia University and at the University of Colorado-Boulder.

References

ARDA and NIST. 2003. *AQUAINT R&D Program 18 Month Workshop*, San Diego, CA.

Sasha Blair-Goldensohn, Kathleen R. McKeown, and Andrew Hazen Schlaikjer. 2003. A hybrid approach for answering definitional questions. Technical Report CUCS-006-03.

William W. Cohen. 1995. Fast effective rule induction. In *Proc. of 12th Int'l Conf. on Machine Learning*, pages 115–123.

Pablo A. Duboue and Kathleen R. McKeown. 2003. Statistical acquisition of content selection rules for natural language generation. In *EMNLP 2003*.

Ralph Grishman. 1997. Information extraction: Techniques and challenges. In *SCIE*, pages 10–27.

Eduard H. Hovy. 1993. Automated discourse generation using discourse structure relations. *Artificial Intelligence*, 63(1-2):341–385.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Daniel Marcu. 1997. Rhetorical parsing of natural language texts. In *In Proc. ACL–EACL 97*, pages 96–103.

Kathleen R. McKeown. 1985. *Text Generation*. Cambridge Univ. Press.

Johanna D. Moore and Cecile L. Paris. 1992. Planning text for advisory dialogues: Capturing intentional and rhetorical information. *Comp Ling*, 19(4):651–695.

Dragomir R. Radev and Kathleen R. McKeown. 1998. Generating natural language summaries from multiple on-line sources. *Comp Ling*, 24(3):469–500.

E. Reiter and R. Dale, 2000. *Building Natural Language Generation Systems*, chapter 4. Cambridge Univ. Press.

Juan C. Sager and M.C. L'Homme. 1994. A model for definition of concepts. *Terminology*, pages 351–374.

Margaret Sarner and Sandra Carberry. 1988. A new strategy for providing definitions in task oriented dialogues. In *Proc. Int'l Conf. on Computational Linguistics (COLING-88)*, volume 1.

Robert E. Schapire and Yoram Singer. 2000. BoosTexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168.

Norman Swartz. 1997. Definitions, dictionaries and meanings. Available online http://www.sfu.ca/philosophy/definitn.htm.

Ellen Voorhees. 2002. Overview of the trec 2002 question answering track. In *Text Retrieval Conference 2002*, Gaithersburg, MD. NIST.

Ellen Voorhees. 2003. Answers to definition questions. In *HLT-NAACL 2003*, pages 109–111, Edmonton, Canada.