# Corpus-based Generation of Easy Help-desk Responses

Yuval Marom and Ingrid Zukerman
School of Computer Science and Software Engineering
Monash University
Clayton, VICTORIA 3800, AUSTRALIA
{yuvalm,ingrid}@csse.monash.edu.au

**Abstract**

We present a corpus-based approach for the automatic generation of email responses to help-desk requests. This is largely an extractive multi-document summarization task. However, in our application users have a very low tolerance for responses that contain incongruous sentences. To address this problem, we propose a method for extracting high-precision sentences for inclusion in a response, and a measure for predicting the completeness of a planned response. The idea is that complete, high-precision responses may be sent to users, while incomplete responses should be passed to operators. Our results show that a small but significant proportion (14%) of our automatically generated responses have a high degree of precision and completeness, and that our measure can reliably predict the completeness of a response.

## 1 Introduction

People often email help-desks to solve problems pertaining to services or goods provided by organizations. Organizations respond to this demand by allocating operators whose job is to respond to these requests, and by developing "how to" manuals to enable these operators to provide consistent and helpful responses. This solution is expensive, and as a result the supply of operators typically falls short of the demand, thereby resulting in a reduced standard of service. Further, even though many of the submitted queries are repetitive, the operators still have to spend time addressing them, which takes them away from more complex requests.

In this paper, we present an initial report of our corpus-based approach to alleviate this problem with respect to email inquiries sent to Hewlett Packard (HP). Our objective is to automatically generate responses to users' requests on the basis of similar responses seen in a corpus of email dialogues. This is essentially an extractive multi-document summarization task, in that similar documents (email responses) are first identified, followed by the automatic extraction of important sentences. However, there is an important difference between our task and traditional multi-document summarization. Normally, the inclusion of an irrelevant information item in a summary does not invalidate the summary. In contrast, in our application, a response email

| **Request:** Return label was not under the shipping tag and I have been waiting nearly two weeks for a label after reporting it not attached to the box. | **Request:** Hi There, I acquired an *ORG MODEL* Tape Drive from a friend and would like to know how I go about setting it up for use with WinXP. XP does not seem to detect the drive at all. HELP? |
|---|---|
| **Complete response:** I apologize for the delay in responding to your issue. Your request for a return airbill has been received and has been sent for processing. Your replacement airbill will be sent to you via email within 24 hours. | **Incomplete response:** Thank you for contacting *ORG*'s Customer Care Technical Center. We are only able to assist customers with in warranty products through our email services. At the present time, we have the following numbers to contact technical support for your out of warranty product. *Please call PHONENUM. This facility will be available from Monday to Friday between 9.00 AM to 5.00 PM. For additional information, please visit the link given below: WEBSITE.* |

Figure 1: Request with a complete response (left column) and request with an incomplete response (right column)

that contains even one incongruous sentence may alienate a user. As a result, the responses generated by our system must have very high relevance (often at the expense of completeness).

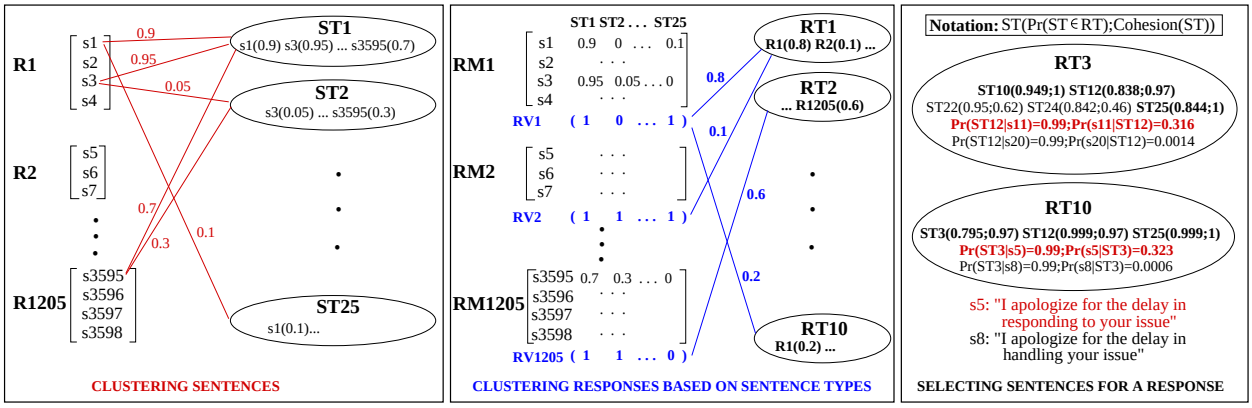
To generate such responses, we have developed a procedure that selects high-precision sentences from a cluster of similar responses, and a measure that predicts the completeness of the resultant responses from the features of their source cluster. The idea is that high-precision responses with a high predicted degree of completeness may be sent directly to users, while incomplete responses should be passed to an operator. For example, the left-hand column in Figure 1 shows a request and a complete response automatically generated by our system; the right-hand column shows a request and an incomplete response (the additional information in the operator's response and the extra plural in "numbers" in our system's response have been italicized).

Our results show that the completeness of the responses often depends on the topic of the user's request, that a small but significant proportion (14%) of our automatically generated responses have a high degree of completeness, and that our measure can reliably predict the completeness of a response.

Section 2 outlines our domain and corpus. Our response-generation process is described in Section 3, and the results of our evaluation in Section 4. Section 5 discusses related work, followed by concluding remarks.

# 2 Domain

Our corpus consists of 30000 email dialogues between users and help-desk operators at HP. These dialogues deal with a variety of user requests, which include requests for technical assistance, inquiries about products, and queries about how to return faulty products or parts.




(a) Sentence clustering (b) Response clustering (c) Response generation

Figure 2: Sentence and response clustering, and response generation

The subject line for each request includes an HP topic selected by the user from a menu in the "contact-us" interface. This topic is significant, as there is little commonality between the requests (and their replies) for different topics. We therefore clustered the email dialogues using as features the lemmatized words in the subject line, including the HP topic (a lemma is the uninflected version of a word). This was done using the SNOB program [Wallace and Boulton, 1968], which yielded 38 subject-based clusters that contain between 25 and 8000 email dialogues (SNOB automatically determines the number of clusters). Examples are: "product replacement" (PRDRP) with 1416 dialogues, and "desktop" (DESKTOP) with 590 dialogues.[1] In the rest of this paper, the subject-based clusters are treated as separate datasets.

We then re-applied SNOB to cluster the dialogues inside each dataset using three dialogue features: *number of turns* (emails), *length* (number of email lines), and *duration* (number of days until dialogue completion). *Number of turns* and *duration* are the most significant features, yielding large sub-clusters of two-turn dialogues that last one day or less. For instance, in the PRDRP dataset 85% of the dialogues are in this sub-cluster, and in the DESKTOP dataset 91% of the dialogues. This pattern is consistent across all the subjects, with an average of 80% of the dialogues having two turns: request and reply. These two-turn dialogues are the focus of our work, as users are seemingly satisfied with the operators' replies. The hope is that similar replies may be automatically generated.

# 3 Procedure

Our response-generation method consists of the following main steps: (1) identifying types of sentences; (2) clustering email responses according to the sentence types they contain; (3) calculating the "semantic compactness" of the response clusters; and (4) selecting sentences for inclusion in a response (Figure 2 illustrates Steps 1, 2 and 4).[2]

[1] SNOB produces numbered clusters. The names of the clusters were created by the authors.

[2] Owing to time limitations, the procedures described in this section were applied to 40% of the data, which contains datasets comprising between 300 and 1500 2-turn/1-day-or-less dialogues.

## 3.1 Identifying types of sentences

We apply SNOB to cluster the sentences in each help-desk response into *Sentence Types (STs)*. The features used for clustering are the lemmatized words in the sentences. Feature selection is performed separately for each dataset, yielding a different bag of words for each dataset. For instance, the PRDRP dataset yields 76 lemmatized words, and the DESKTOP dataset 556 words. Each sentence is then represented by means of a binary vector of size $N$ (number of feature words in the dataset), where element $j$ is 1 if word $w_j$ is present in the sentence, and 0 otherwise.

SNOB yields $m$ sentence types, where $m$ varies for each dataset. For example, the PRDRP dataset has 25 sentence types, and the DESKTOP dataset has 40. Each sentence type $ST_i$, $i = 1, \ldots, m$, is represented by means of a centroid $CST_i$ — an $N$-dimensional vector, such that $CST_i[j] = \Pr(w_j \in ST_i)$ (the probability that word $w_j$ is used in $ST_i$). Figure 2(a) illustrates the sentence-clustering process for the PRDRP dataset, which contains 1205 responses comprising a total of 3598 sentences. As seen in this example, a sentence may probabilistically belong to more than one sentence type, e.g., $\Pr(ST1|s1) = 0.9$ and $\Pr(ST25|s1) = 0.1$ (these probabilities are returned by SNOB).

## 3.2 Clustering help-desk responses

We apply SNOB again to cluster help-desk responses into *Response Types (RTs)*, but first we perform the following steps to represent responses by means of sentence types.[3]

**Representing sentences in terms of sentence types.** We represent each sentence $s_j$ by means of an $m$-dimensional vector, where $m$ is the number of sentence types. Element $i$ in the vector for sentence $s_j$ contains $\Pr(ST_i|s_j)$ (the probability that $s_j$ belongs to sentence type $ST_i$). We then combine the vector for each sentence in response $R_k$ into a *Response Matrix* $RM_k$ of size $n_k \times m$, where $n_k$ is the number of sentences in $R_k$. For instance, as seen in Figure 2(b), RM1, the response matrix for response R1, comprises the vectors for sentences s1, s2, s3 and s4; the vector for s1 indicates that $\Pr(ST1|s1) = 0.9$, $\Pr(ST25|s1) = 0.1$ and $\Pr(ST_j|s1) = 0$ for $j = 2, \ldots, 24$ (these probabilities sum to 1).

**Representing help-desk responses in terms of sentence types.** For each response matrix $RM_k$, we derive an $m$-dimensional *Response Vector* $RV_k$, such that for $i = 1, \ldots, m$

$$RV_k[i] = \begin{cases} 1 & \text{if } \exists RM_k[j,i] \geq 0.1 \text{ for } j = 1, \ldots, n_k \\ 0 & \text{otherwise} \end{cases}$$

That is, $RV_k[i] = 1$ indicates that sentence type $ST_i$ has some presence in response $R_k$ (with probability $\geq 0.1$). The discrimination between sentence types with varying degrees of presence in $R_k$ is performed at a later stage (Section 3.4).

[3] In our initial experiments, we tried to cluster the responses directly from a bag-of-words as done in [Radev et al., 2000]. However, this approach did not yield useful clusters, because it finds similarities between documents, rather than between parts of documents.

**Clustering.** The response vectors are given to SNOB, which clusters them into response types. The number of response types varies for different datasets. For instance, PRDRP and DESKTOP have 10 and 9 response types respectively. Each response type $RT_l$ is represented by means of a centroid $CRT_l$ — an $m$-dimensional vector, such that $CRT_l[i] = \Pr(ST_i \in RT_l)$ (the probability that sentence type $ST_i$ is used in response type $RT_l$).

## 3.3 Calculating the semantic compactness of a response type

The *Semantic Compactness* (*SemCom*) of a response type is a measure that predicts whether it is possible to generate a complete, high-precision response from this response type. This measure calculates the proportion of highly cohesive and frequent sentence types among the sentence types that have some presence in the response type. If this proportion is high, the response type is deemed semantically compact, which means that it is a good candidate for automatic response generation. As the value of this proportion decreases, so does the likelihood of automatically generating a complete response from the response type in question. During response generation, our system compares the semantic compactness of a response type with an empirically determined threshold, in order to determine whether an operator should participate in the composition of a reply. In Section 4, we evaluate the semantic compactness measure, and suggest a value for its threshold.

Formally, we define the semantic compactness of a response type $RT_l$ as follows.

$$\mathit{SemCom}(RT_l) = \frac{\sum_{i=1}^{m} \delta_{Hi}(RT_l, ST_i)}{\sum_{i=1}^{m} \delta_{Low}(RT_l, ST_i)} \tag{1}$$

where $m$ is the number of sentence types,

$$\delta_{Low}(RT_l, ST_i) = \begin{cases} 1 & \text{if } \Pr(ST_i \in RT_l) \geq 0.1 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$\delta_{Hi}(RT_l, ST_i) = \begin{cases} 1 & \text{if } [\ \Pr(ST_i \in RT_l) \geq \mathcal{T}_{Hi} \wedge \\ & \quad \mathit{Cohesion}(ST_i) \geq \mathcal{T}_{Coh}\ ] \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

In turn, $\mathit{Cohesion}(ST_i) = \frac{1}{N} \sum_{j=1}^{N} \delta_{\alpha}(ST_i, w_j)$, where $N$ (the number of words in the dataset) is the dimension of $CST_i$ (the centroid for sentence type $ST_i$), and

$$\delta_{\alpha}(ST_i, w_j) = \begin{cases} 1 & \text{if } [\ \Pr(w_j \in ST_i) \leq \alpha \vee \\ & \quad \Pr(w_j \in ST_i) \geq 1 - \alpha\ ] \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Finally, $\mathcal{T}_{Hi}$, $\mathcal{T}_{Coh}$ and $\alpha$ are empirically determined thresholds (Section 4.3).

The denominator of Equation 1 counts the number of sentence types that appear in response type $RT_l$ with some frequency (hence the low threshold of 0.1 in Equation 2). The numerator counts the number of highly cohesive sentence types that appear in $RT_l$ very frequently. The cohesion of a sentence type is the proportion of words that are almost certainly present or almost

certainly absent from this sentence type. The rationale for this measure is that a cohesive group of sentences should agree strongly on the words they use and the words they omit. Thus, cohesive sentence types usually comprise sentences that are very similar to each other, while loose sentence types comprise dissimilar sentences. Hence, it is possible to obtain a sentence that adequately represents a cohesive cluster, while this is not the case for a loose cluster.

In the experiments reported in Section 4, our thresholds have rather stringent values ($\mathcal{T}_{Hi} = 0.75$, $\mathcal{T}_{Coh} = 0.9$ and $\alpha = 0.01$), in order to implement a cautious approach that avoids including potentially incongruous sentences in automatically generated responses. However, our sensitivity analysis shows that the quality of our responses is largely maintained even if we relax some of these thresholds (Section 4.3).

## 3.4 Selecting sentences for inclusion in a response

In order to select sentences for inclusion in a response, we use a modified version of the *adaptive greedy algorithm* proposed in [Filatova and Hatzivassiloglou, 2004]. This algorithm, which was used for extractive multi-document summarization, calculates the score for each sentence in the documents as the sum of the weights of the concepts it covers. The sentence with the highest score is then included in the output, and the weights of the concepts covered by this sentence are subtracted from the scores of the other sentences that also cover these concepts.

Our modifications pertain to the treatment of sentence types as concepts, and the probabilistic association between sentences and sentence types (as in the sentence vectors in Figure 2(b)). These modifications entail the following formula for calculating the score of a sentence $s_j$.

$$\textit{Score}(s_j) \;=\; \sum_{i=1}^{m} \{\Pr(s_j|ST_i) \times \Pr(ST_i \in RT_l) \times \delta_{Coh}(ST_i)\} \tag{5}$$

where $\Pr(s_j|ST_i)$ and $\Pr(ST_i \in RT_l)$ are returned by SNOB,[4] and

$$\delta_{Coh}(ST_i) = \begin{cases} 1 & \text{if } \textit{Cohesion}(ST_i) \geq \mathcal{T}_{Coh} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

Equation 5 assigns a high score to sentences that are representative of sentence types that (a) have a high probability of appearing in the response type in question, and (b) are highly cohesive (as mentioned in Section 3.1, the cohesion of a sentence type indicates how easy it is to find in it representative sentences). Further, the formula strongly penalizes sentences that belong to non-cohesive sentence types, in order to ensure high precision in the response.

Figure 2(c) illustrates the generation of a response for two response types, RT3 and RT10 (for example, the notation ST12(0.838;0.97) means that $\Pr(ST12 \in RT3) = 0.838$ and that $\textit{Cohesion}(ST12) = 0.97$). RT3 contains five sentence types with a high value for $\Pr(ST_i \in RT3)$, but only three of them have high cohesion (ST10, ST12 and ST25). Hence, we only include sentences from these sentence types in the summary for RT3. In contrast, all the sentence types in RT10 have a high value for $\Pr(ST_i \in RT10)$ and high cohesion. Hence, a complete summary can be generated from these sentence types. As seen in Equation 5, since the values for

[4] $\Pr(s_j|ST_i)$, which reflects how representative sentence $s_j$ is of sentence type $ST_i$, is different from $\Pr(ST_i|s_j)$, the probability that $s_j$ belongs to sentence type $ST_i$ (shown in Figures 2(a) and 2(b)).

Pr($ST_i \in RT10$) and *Cohesion*($ST_i$) are high for all the sentence types in RT10, the selection of a representative sentence for these sentence types depends mainly on Pr($s_j|ST_i$). In this example, s5 is selected to represent ST3, as Pr($s5|ST3$) is much higher than Pr($s8|ST3$). The resultant response is that in the left-hand column of Figure 1.

# 4 Evaluation

In this section, we assess the predictive power of our semantic compactness measure, and the ability of our procedure to generate high-precision responses with a high level of completeness.

## 4.1 Methodology

Our *SemCom* measure is designed to predict the completeness of an automatically-generated response composed of high-precision sentences. In order to determine the utility of this measure, we examine how well it correlates with the quality of the generated responses.

We assess the quality of a generated response $r_g$ by comparing it with the actual responses in the response type from which $r_g$ was sourced. To this effect, we use three well-known IR measures: precision, recall and F-score [Salton and McGill, 1983]. Precision gives the proportion of words in $r_g$ that match those in an actual response; recall gives the proportion of words in the actual response that are included in $r_g$; and F-score is the geometric average of precision and recall. Precision, recall and F-score are then averaged over the responses in $r_g$'s response type to give an overall evaluation of $r_g$.[5]

## 4.2 Example — The PRDRP dataset

Table 1 lists the response types generated for the PRDRP dataset (response types RT3 and RT10 also appear in Figure 2(c)). For each response type, it lists the number of responses represented by the response type; the *SemCom* measure; and the average precision, recall and F-score of the generated response. From this table we see that precision is generally high, and is uncorrelated with *SemCom*. This is not surprising, as the sentence-selection process is designed to select high-precision sentences. Hence, so long as at least one sentence is selected (which is not the case for response types RT7 and RT9), the text in the generated response $r_g$ will agree with the text in the responses that are represented in $r_g$'s response type. In contrast, recall is highly correlated with *SemCom*. A decrease in *SemCom* indicates that fewer sentences are included in the generated response, which therefore covers less of the information in the original responses. As expected from these results, the overall F-score is also highly correlated with semantic compactness. Finally, it is worth noting that this dataset contains three response types with maximum semantic compactness (RT1, RT6 and RT10), which have a very high F-score (although RT1 has a slightly lower recall). This means that 81.5% of the responses in this dataset may be completely generated by our system.

[5] In addition to these measures, which are calculated on a word-by-word basis, we considered the ROUGE evaluation procedure, which also takes into account word sequences [Lin and Hovy, 2003]. However, since in our application the simpler word-by-word evaluation correlates well with ROUGE, we report on the former.

Table 1: Semantic compactness of response types, and evaluation of generated responses for the PRDRP dataset

| RT | # of Responses | *SemCom* | Average Precision | Average Recall | Average F-score |
|---|---|---|---|---|---|
| 1 | 72 | 1.00 | 0.84 | 0.73 | 0.78 |
| 2 | 67 | 0.75 | 0.83 | 0.78 | 0.80 |
| 3 | 11 | 0.43 | 0.85 | 0.40 | 0.54 |
| 4 | 31 | 0.50 | 0.73 | 0.41 | 0.52 |
| 5 | 20 | 0.66 | 0.80 | 0.52 | 0.62 |
| 6 | 49 | 1.00 | 0.82 | 0.82 | 0.82 |
| 7 | 30 | 0.00 | 0.00 | 0.00 | 0.00 |
| 8 | 4 | 0.33 | 0.92 | 0.34 | 0.49 |
| 9 | 55 | 0.00 | 0.00 | 0.00 | 0.00 |
| 10 | 862 | 1.00 | 0.81 | 0.80 | 0.80 |

## 4.3 Results

**Predictive performance of *SemCom*.** Figure 3 shows the relationship between semantic compactness and precision, recall and F-score for the 135 response types created for the different datasets. We can see from this figure that the precision and recall characteristics observed for the PRDRP dataset in Table 1 generalize to all the datasets. Namely, precision is high and is uncorrelated with *SemCom*, while recall and hence F-score are strongly correlated with *SemCom*. Figure 3 also suggests a threshold of 0.7 to indicate high semantic compactness, and a further threshold of 0.4 to indicate medium semantic compactness. The idea is that these thresholds will assist in the selection a response generation strategy for a response type (see *Overall system performance*).

In order to confirm the agreement between semantic compactness and the IR measures, we calculated the statistical correlation between *SemCom* and F-score (both linear and log correlation). The results in Figure 3 yield a linear correlation of 0.89 and a log correlation of 0.9, which demonstrate the high predictive power of the *SemCom* measure. However, for the predictions made by *SemCom* to be useful, they must also agree with users' views. To test whether this is the case, we conducted a small, preliminary study as follows. We constructed four evaluation sets by selecting four response types with high semantic compactness ($\geq 0.7$), automatically generating a response from each response type, and selecting 15 actual responses from each response type for comparison.[6] Each evaluation set was given to two judges, who were asked to rate the precision and completeness of the generated response compared to each of the 15 responses in the set. Our judges gave all the automatically generated responses high precision ratings, and completeness ratings which were consistent with our semantic compactness measure.

**Sensitivity analysis of parameters.** The results in Figure 3 were obtained with parameter values $\alpha = 0.01$, $\mathcal{T}_{Hi}$=0.75 and $\mathcal{T}_{Coh}$=0.9 in Equations 3 and 4. The most sensitive of these parameters is $\alpha$, which means that words have to agree quite strongly for a group of similar sentences to be considered cohesive. $\mathcal{T}_{Coh}$ is less sensitive, particularly for values greater than

[6]Several of our automatically-generated responses match perfectly the operators' responses. Since these are obvious matches, they were not included in our study.

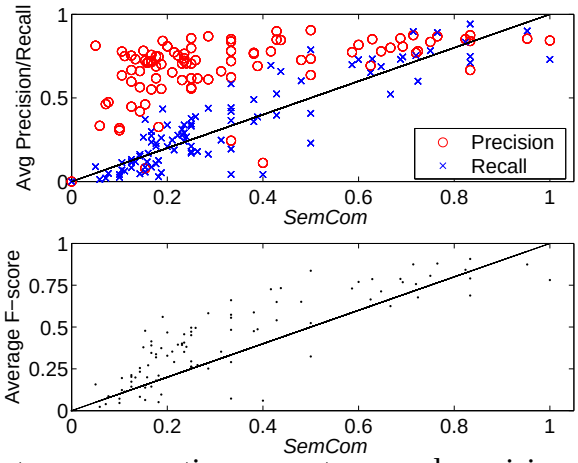


Figure 3: Relationship between semantic compactness and precision and recall (top plot), and F-score (bottom plot)

0.8, i.e., at least 80% of the words in a sentence type must pass the cohesion test for our measure to be reliable. The least sensitive parameter is $\mathcal{T}_{Hi}$. This is not surprising when we consider that this parameter represents the agreement between the responses in a response type on their usage of sentence types — a certain level of agreement is guaranteed by the clustering process, which groups similar responses on the basis of sentence types. In other words, a group of responses will either agree or disagree about using a particular sentence type, i.e., Pr$(ST_i \in RT_l)$ will be quite low or quite high, and hence the value of the threshold is not critical.

**Overall system performance.** The overall performance of our system was measured in terms of the proportion of high-precision, complete responses that can be generated from our corpus without human intervention. These are the responses that are represented by response types with high semantic compactness. As mentioned above, Figure 3 suggests a threshold of 0.7 for high semantic compactness. That is, responses that are generated from response types that exceed this threshold could be directly sent to users. This would result in the automatic remittance of approximately 14% of the responses. The application of the medium semantic compactness threshold of 0.4 would result in a further 6% of the generated responses being passed to an operator. The remaining 80% of the responses would have to be mostly written by an operator. However, this may be a pessimistic estimate, as some response types with a low *SemCom* yield reasonable responses, such as that in the right-hand column of Figure 1, which has a *SemCom* of 0.25. It is also worth noting that the above percentages vary across the different datasets. Some sets, such as PRDRP and TAPEDRV, have a significant percentage of responses that belong to a response type with high semantic compactness. In contrast, datasets such as DESKTOP and PORTBL, have no such responses. This observation indicates that it may be fruitful to focus the automatic response-generation effort on particular topics.

# 5 Related Research

The idea of clustering text and then generating a summary from the clusters has been implemented in previous multi-document summarization systems. Radev *et al.* (2000) employed a

centroid-based approach, where documents are clustered using a bag-of-words representation. They summarized each cluster by scoring sentences based on their closeness to the centroid of the cluster, and including the highest-scoring sentences in the summary. Hatzivassiloglou *et al.* (2001) used a tagged corpus to train a (supervised) classifier to determine the similarity between paragraphs. They used word-based features as well as higher-level features such as word overlap, proper noun overlap and noun phrase head. The similarity between paragraphs was used to cluster the paragraphs, and the selection of paragraphs for inclusion in a summary favoured those from clusters that represent many documents.

Filatova and Hatzivassiloglou (2004) proposed a more general approach to sentence selection, where textual units of arbitrary length are associated with "concepts". They considered several algorithms for selecting sentences which "cover" as many concepts as possible while avoiding redundancy. In principle, these concepts could be word-based features or higher-level features. In their implementation, concepts are tuples composed of verbs and nouns that appear frequently in the corpus. Thus, a sentence is deemed to cover a concept if it mentions the nouns and verbs in this concept. Such concepts are useful for summarizing news articles, because the noun-verb combinations often capture the gist of an article.

Our work employs this general model, and particularly Filatova and Hatzivassiloglou's sentence selection algorithm. However, their use of concepts is not suitable for our application, where the wording of a reply is crucial. For this reason, our "concepts" are created directly from the actual sentences in the corpus, which are clustered to form sentence types.

Thus, our work differs from previous work on clustering and summarization in two respects. Firstly, the high-level features (sentence types) we use to cluster documents are learned from the corpus in an unsupervised manner, using as input only low-level, word-based features. Secondly, our reliance on sentence types enables us to identify response patterns beyond those identified by topic words, and hence allows us to generate different types of summaries within a single topic.

The related work mentioned above, like most of the work on multi-document summarization, is applied to news articles. Two applications which summarize emails are described in [Corston-Oliver et al., 2004, Dalli et al., 2004]. However, these systems perform single-document summarization, where a thread of emails is reformulated to highlight the key issues in the discussion. Although this task is rather different from ours, some important issues arise from this work, particularly concerning the use of high-level features that are suitable for emails. Specifically, Corston-Oliver *et al.* trained a (supervised) classifier to detect "email acts" from a tagged corpus, and reported that such features are superior to word-based features for their summarization system.

# 6 Conclusion

We have offered a corpus-based approach for the automatic generation of responses to help-desk requests — a task where users exhibit a very low tolerance to irrelevant information. Our approach, which uses an unsupervised learning perspective in combination with a simplistic bag-of-words representation, has enabled our system to generate a small but significant proportion (14%) of the email responses in our corpus. Specifically, our main contributions are:

- A multi-document summarization system that clusters responses (documents) based on sentence types, which reflect similarities between parts of responses. This enables our system to identify different types of responses within a single topic, and generate a response that "summarizes" each response type.
- A semantic compactness measure that reliably predicts the completeness of a high-precision response, and that can be used to select a response-generation strategy.
- An adaptation of the sentence-selection procedure described by Filatova and Hatzivassiloglou (2004) to select high-precision sentences on the basis of sentence types.

Our results also show that the majority of the responses (80%) are quite specific, and require substantial operator intervention according to our semantic compactness measure. However, we believe that with a more powerful approach, further advances are possible for automating these responses. To this effect, we propose to do the following.

- Adopt a semantically-oriented sentence representation. This will require the use of a context-dependent word-similarity measure, such as that proposed by Lin (1998).
- Perform linguistic analysis to extract higher-level discourse features, and apply machine learning techniques to extract pragmatic features, such as email acts.
- Learn grammars of responses and sentence types. This would support the identification and generation of common sentence sequences, such as steps in a procedure, and hopefully higher-level structures.

Finally, our current response-generation process assumes that a user's request can be matched to a response type. Hence, this process is mainly based on the content of the responses. In the near future, we will implement a query-relevant version of this process where the sentences in a user's request are used to prime (increase the probability of) related sentence types.

# References

[Corston-Oliver et al., 2004] Corston-Oliver, S., Ringger, E., Gamon, M., and Campbell, R. (2004). Task-focused summarization of email. In *Proceedings of the ACL 2004 Workshop Text Summarization Branches Out*, pages 43–50, Barcelona, Spain.

[Dalli et al., 2004] Dalli, A., Xia, Y., and Wilks, Y. (2004). Adaptive information management: FASiL email summarization system. In *COLING'04 – Proceedings of the 20th International Conference on Computational Linguistics*, pages 23–27, Geneva, Switzerland.

[Filatova and Hatzivassiloglou, 2004] Filatova, E. and Hatzivassiloglou, V. (2004). Event-based extractive summarization. In *Proceedings of ACL'04 Workshop on Summarization*, pages 104–111, Barcelona, Spain.

[Hatzivassiloglou et al., 2001] Hatzivassiloglou, V., Klavans, J., Holcombe, M., Barzilay, R., Kan, M., and McKeown, K. (2001). Simfinder: A flexible clustering tool for summarization. In *Proceedings of the NAACL Workshop on Automatic Summarization*, Pittsburgh, Pennsylvania.

[Lin and Hovy, 2003] Lin, C. and Hovy, E. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Language Technology Conference (HLT-NAACL 2003)*, Edmonton, Canada.

[Lin, 1998] Lin, D. (1998). Automatic retrieval and clustering of similar words. In *COLING-ACL'98 – Proceedings of the International Conference on Computational Linguistics and the Annual Meeting of the Association for Computational Linguistics*, pages 768–774, Montreal, Canada.

[Radev et al., 2000] Radev, D., Jing, H., and Budzikowska, M. (2000). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *ANLP/NAACL2000 Workshop on Summarization*, Seattle, Washington.

[Salton and McGill, 1983] Salton, G. and McGill, M. (1983). *An Introduction to Modern Information Retrieval*. McGraw Hill.

[Wallace and Boulton, 1968] Wallace, C. and Boulton, D. (1968). An information measure for classification. *The Computer Journal*, 11(2):185–194.