A Learning-based Sampling Approach to Extractive Summarization

Vishal Juneja and Sebastian Germesin and Thomas Kleinbauer
German Research Center for Artificial Intelligence
Campus D3.2
66123 Saarbücken, Germany
{firstname.lastname}@dfki.de

Abstract

In this paper we present a novel resampling model for extractive meeting summarization. With resampling based on the output of a baseline classifier, our method outperforms previous research in the field. Further, we compare an existing resampling technique with our model. We report on an extensive series of experiments on a large meeting corpus which leads to classification improvement in weighted precision and f-score.

1 Introduction

Feature-based machine learning approaches have become a standard technique in the field of extractive summarization wherein the most important sections within a meeting transcripts need to be identified. We perceive the problem as recognizing the most extract-worthy meeting dialog acts (DAs) in a binary classification framework.

In this paper, firstly, in section 4 we create a *gold standard* to train the classifier, by improvising upon the existing annotations in our meeting corpus. Then in section 5 we present actual numbers which display a very skewed class distribution to learn for the binary classifier. This skewness is attributed to the less number of actual extract-worthy and important DAs (positive examples) compared to ordinary chit-chat, backchannel noises etc (negative examples) spoken during the course of the meeting. We tackle this data skewness with a novel resampling approach which reselects the data set to create a more comparable class distribution between these postive and negative instances.

Resampling methods have been found effective in catering to the data imbalance problem mentioned above. (Corbett and Copestake, 2008) used a resampling module for chemical named entity recognition. The pre-classifier, based on n-gram character features, assigned a probability of being a chemical word, to each token. Only tokens having probability greater than a predefined threshold were preserved and the output of the first stage classification along with word suffix were used as features in further classification steps. (Hinrichs et al., 2005) used a hybrid approach for Computational Anaphora Resolution (CAR) combining rule based filtering with Memory based learning to reduce the huge population of anaphora/candidate-antecedent pairs. (Xie et al., 2008), in their experimentation on the ICSI meeting corpus, employ the salience scores generated by a TFIDF classifier in the resampling task. We discuss the actual technique and our resampling module further in section 6.

We compare its performance with the TFIDF model of (Xie et al., 2008) in section 8.2 and observe a general improvement in summary scores through resampling.

2 Data

We use the scenario meetings of the AMI corpus for our experiments in this paper which comprise about two thirds of around 100 hours of recorded and annotated meetings. The scenario meetings each have four participants who play different roles in a fictitious company for designing a remote control. The AMI corpus has a standard training set of 94

Proceedings of the NAACL HLT 2010 Student Research Workshop, pages 34–39,
Los Angeles, California, June 2010.

meetings[1] and 20 meetings each for development and testing.

Annotators wrote abstractive summaries for each meeting and then linked summary sentences to those DA segments from the meeting transcripts which best conveyed the information in the abstracts. There was no limit on the number of links an annotator could create and a many-to-many mapping exists between the meeting DA segments and human abstracts. Here, DA segments are used in analogy to sentences in document summarization because the spontaneously spoken material in meeting transcripts rarely contains actual grammatical sentences.

3 Pre-processing and Feature Extraction

To the feature set of (Murray, 2008) listed in table 1 we add some high level features. Since the main focus of this paper is to deal with the data imbalanace issue hence for the sake of completeness and reproducibility of our work we briefly mention the basic features used. In section 8.3 we explicitly report the performance rise over the baseline due to the added features.

3.1 Lexical and Structural features

The list of added features include the number of content words (nouns and adjectives) in a DA. (Edmundson, 1969) looked at cue-phrases, keywords title and location of a sentence as features indicative of important sections in a document. We use a handpicked list of cue words like "for example", "gonna have" etc as binary features. We also add several keywords like "remote","plastic" etc based upon manual scrutiny, as binary features into the classifier. Further we use DA labels of current and four adjacent DAs as features.

3.2 Disfluency

The role of disfluencies in summarization has been investigated by (Zhu and Penn, 2006) before. They found that disfluencies improve summarization performance when used as an additional feature. We count the number of disfluent words in a DA using an automatic disfluency detector.

[1]Three of the meetings were missing some required features.

3.3 Prosodic

We employ all the signal level features described by (Murray, 2008) which include mean, max and standard deviation of energy and pitch values normalized by both speaker and meeting. The duration of the DA in terms of time and number of words spoken. The subsequent, precedent pauses and rate of speech feature.

DA Features
mean energy
mean pitch
maximum energy value
maximum pitch value
standard deviation of pitch
precedent pause
subsequent pause
uninterrupted length
number of words
position in the meeting
position in the speaker turn
DA time duration
speaker dominance in DA
speaker dominance in time
rate of speech
SUIDF score
TFIDF score

Table 1: Features used in baseline classifier

4 Gold Standard

In supervised frameworks, the creation of *gold-standard* annotations for training (and testing) is known to be a difficult task, since (a) what should go into a summary can be a matter of opinion and (b) multiple sentences from the original document may express similar content, making each of them equally good candidates for selection. The hypothesis is well supported by the low *kappa* value (Cohen, 1960) of 0.48 reported by (Murray, 2008) on the AMI corpus.

We describe the procedure for creating the gold standard for our experimentation in this paper. Firstly we join all annotations and rank the DAs from most number of links to least number of links to create a sorted list of DAs. Depending on a predefined variable percentage as gold standard cut-off

or threshold we preserve the corresponding number of highest ranked DAs in the above list. For evaluation, (Murray, 2008) uses gold standard summaries obtained using similar procedure. For training, however, he uses all DA segments with at least one link as positive examples.

As the term gold standard for the data set, created above, is misleading. We call the set of DAs so obtained by using this ranking and resampling procedure as Weighted-Resampled Gold Standard (WRGS). Henceforth in this paper, for a resampling rate of say 35% we will name the set of DAs so obtained as WRGS(35%) or simply WRGS for some undefined, arbitrary threshold.

5 Data Skewness

In this section we focus on the skewed data set which arises because of creating WRGS for training our classifiers. Consider the set of DAs with at least one link to the abstractive or human summaries. Let us call it $DA^{l\geq 1}$. This set accounts for 20.9% of all DAs in the training set.

set	size%
WRGS(25%)	5.22%
$DA^{l\geq 1}$	20.9%

Table 2: Set sizes in % of all training DAs

Again consider set of DAs for WRGS(25%). This set, by definition, contains 25% of all DAs in the set $DA^{l\geq 1}$. Hence the set WRGS(25%) constitute 5.22% of all DAs in the training set. Note that this is a skewed class distribution as also visible in table 2.

Our system employs resampling architecture shown in figure 1. The first classifier is similar in spirit to the one developed in (Murray, 2008) with the additional features listed in section 3. The output we use is not the discrete classification result but rather the probability for each DA segment to be extracted.

These probabilities are used in two ways for training the second classifier: firstly, to create the resampled training set and secondly, as an additional feature for the second classifier. The procedure for resampling is explained in the section 6.

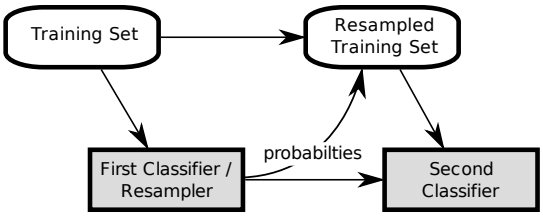

Figure 1: A two-step classification architecture for extractive meeting summarization.

6 Resampling

As explained in previous section our model obtains resampled data for second stage classification using the probabilistic outcomes of a first stage classifier. The resampling is done similar to (Xie et al., 2008) to cater to the data skewness problem. To do the resampling, firstly, the DAs are ranked on decreasing probabilities. In the next step, depending on some resampling rate, a percentage of highest ranked DAs is used in further classification steps, while rest of DA segments are neglected.

(Xie et al., 2008) obtained the resampled set by ranking the DAs on TFIDF weights. Data resampling benefits the model in two ways a) by improving the positive/negative example ratio during the training phase b) by discarding noisy utterances in the test phase as they usually attain low scores from the first classifier.

In testing, the first classifier is run on the test data, its output is used, as in training, to create the resampled test set and the probability features. Finally, the summary is created from the probabilities produced by the second classifier by selecting the highest ranked DA segments for the specified summary length.

As the data for resampling is derived by a learning-based classifier, we call our approach *Learning-Based Sampling* (LBS).

In this paper, we compare our LBS model with the TFIDF sampling approach adopted by (Xie et al., 2008) and present the results of resampling on both models in section 8.2.

For comparison, we use Murray's (2008) state of art extractive summarization model.

7 Evaluation Metric

The main metric we use for evaluating the summaries is the extension of the *weighted precision* evaluation scheme introduced by (Murray, 2008). The measure relies on having multiple annotations for a meeting and a many-to-many mapping discussed in section 2. To calculate weighted precision, the number of times that each extractive summary DA was linked by each annotator is counted and averaged to get a single DA score. The DA scores are then averaged over all DAs in the summary to get the weighted precision score for the entire summary. The total number of links in an extractive summary divided by the total number of links to the abstract as a whole gives the weighted recall score. By this definition, weighted recall can have a maximum score of 1 since it is a fraction of the total links for the entire summary. Also, there is no theoretical maximum for weighted precision as annotators were allowed to create any number of links for a single DA.

Both weighted precision and recall share the same numerator: $num = \Sigma_d \, L_d/N$ where L_d is the number of links for a DA d in the extractive summary, and N is the number of annotators. Weighted precision is equal to $wp = num/D_s$ where D_s is the number of DAs in the extractive summary. Weighted recall is given by $recall = num/(L_t/N)$ where L_t is the total number of links made between DAs and abstract sentences by all annotators, and N is the number of annotators. The f-score is calculated as: $(2 \times wp \times recall)/(wp + recall)$.

In simple terms a DA which might be discussing an important meeting topic e.g. selling price of the remote control etc is more likely to be linked by more than one annotator and possibly more than once by an annotator. Therefore the high scoring DAs are in a way indicative of quintessential topics and agenda points of the meeting. Hence, weighted precision which is number of links per annotator averaged over all the meeting DAs is a figure that aligns itself with average information content per DA in the summary. Low scoring meeting chit-chats will tend to bring the precision score down. We report a weighted precision of 1.33 for 700 word summary extracted using the procedure described in 2 for obtaining gold standard. This is hence a ceiling to the weighted precision score that can be obtained by any summary corresponding to this compression rate. Weighted Recall on the other hand signifies total information content of the meeting. For intelligent systems in general the recall rate increases with increasing summary compression rates while weighted precision decreases[2].

Since we experiment with short summaries that have at most 700 words, we do most of the comparisons in terms of weighted precision values. In the final system evaluation in section 8.3, we include weighted recall and f-score values.

8 Experimental Results and Discussion

8.1 Training on gold standard

Figure 2 shows the weighted precision results on training an SVM classifier with different gold standard thresholds. For example, at a threshold of 60%, the top 60% of the linked DA segments are defined as the gold standard positive examples, all other DA segments of the meeting are defined as negative, non-extraction worthy. The tests are performed on a single stage classifier similar to (Murray, 2008).

In addition, the curves show the behavior of the system at three different summary compression rates (i.e., number of words in the summary). A general tendency that can be observed is the increase in summary scores with decreasing threshold. For 700 word summaries the peak weighted precision score is observed at 35% threshold. The recall rate remains constant as seen by comparing the first two rows of table 5.

We believe that low inter annotator agreement is the major factor responsible for these results. This shows that a reduced subset classification approach will generally improve results when multiple annotations are available.

8.2 Resampling

In this section we compare two resampling models. The TFIDF model explained in section 6 selects best DAs based on their TFIDF scores. As discussed

[2] An important point to notice is that, a high recall rate does not ensure a good content coverage by the summary. As an example, the summary might pick up DAs pertaining to only a few very important points discussed during the meeting which will lead to a high recall rate although lesser important concepts may still be exclusive.

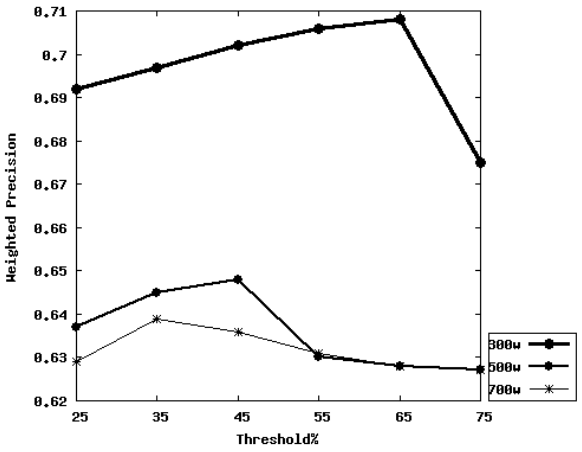

Figure 2: SVM at different compression rates.

previously all sentences above a resampling threshold are preserved while rest are discarded. In 8.2.2 resampling is done from the probabilities of a first stage classifier. SVM model is used for both first and second stage classification.

8.2.1 TFIDF Resampling

Table 3 reports weighted precision and f-scores at two compression rates. The highest f-scores for 700, 1000 word summaries are obtained at 85% and 55% respectively. Plots of figure 3 compare weighted precision scores for LBS and TFIDF models.

# words:	700		1000	
resampl. %	wp	f-score	wp	f-score
15	.631	.217	.600	.274
25	.670	.227	.610	.282
35	.673	.227	.630	.296
55	.685	.231	**.641**	**.305**
75	.689	.232	.632	.302
85	**.692**	**.233**	.631	.299
100	.686	.231	.637	.302

Table 3: TFIDF weighted Precision, f-score for 700 and 1000 word summaries

8.2.2 LBS

The peak performance of the LBS model is observed at resampling rate of 35% for both 700 and 1000 word summaries as seen in table 4. The maximum f-scores, 0.248 and 0.319 (table 4) obtained for LBS outperforms maximum f-scores of 0.233 and 0.305 (table 3) for TFIDF.

# words:	700		1000	
resampl. %	wp	f-score	wp	f-score
15	.684	.236	.662	.309
25	.706	.244	.664	.317
35	**.710**	**.248**	**.664**	**.319**
55	.707	.245	.652	.313
75	.702	.239	.650	.310
85	.702	.239	.642	.307
100	.692	.236	.639	.306

Table 4: weighted precision, f-scores on LBS model

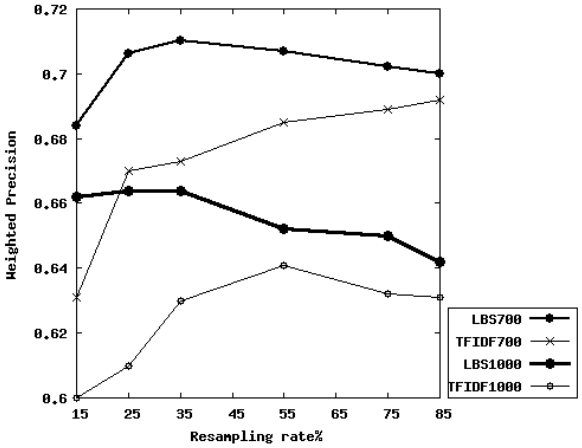

Figure 3: LBS and TFIDF wp values at different compression rates.

From figure 4 which shows positive example retention against sampling rate for TFIDF and LBS it is clear that for all sampling rates, LBS provides a higher rate of positive examples.

Also as discussed above, using a learning-based first classifier produces probability values that can be leveraged as features for the second classifier. We speculate that this also contributes to the differences in overall performance.

8.3 Overall System Performance

In this section we report weighted precision, recall and f-score for 700-word summaries, comparing results of the new model with the initial baseline system.

As shown in table 5, training the system on

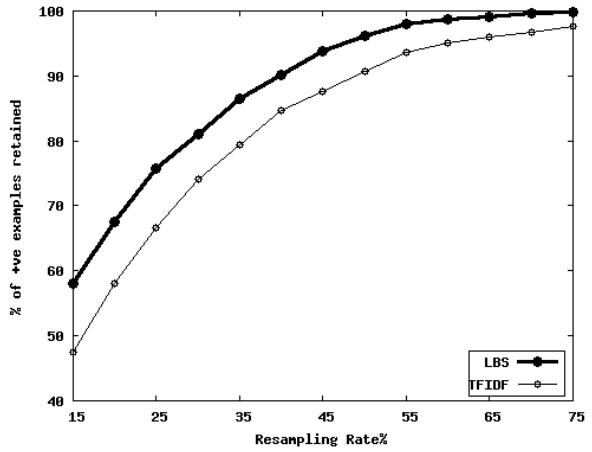

Figure 4: LBS and TFIDF retention rates.

WRGS, with a threshold of 35% increases the precision score from 0.61 to 0.64 while maintaining the recall rate. This is corresponding to the weighted precision score for 35% data point in figure 2.

The last row in table 5 correspond to results obtained with using the LBS proposed in this paper. The scores at 35% resampling are same as the bold faced observations in table 4 for 700 word summaries. We observe that the LBS architecture alone brings about an absolute improvement of 4.41% and 8.69% in weighted precision and f-score.

System	wp	recall	f-score
baseline	0.61	0.13	0.20
+ gold standard	0.64	0.13	0.20
+ new features	0.68	0.15	0.23
+ resampling(LBS 35)%	0.71	0.16	0.25

Table 5: Results on the AMI corpus.

9 Conclusions and Future Work

Through our experimental results in this paper, we firstly observed that training the classifier on WRGS (weighted-resampled gold standard) instances, rather than all the annotated DAs improved the weighted precision scores of our summarizer. We further addressed the problem of skewed class distribution in our data set and introduced a learning-based resampling approach where we resample from the probabilistic outcomes of a first stage classifier. We noted that resampling the data set increased performance, peaking at around 35% sampling rate. We compared the LBS model with the TFIDF resampler obtaining better f-scores from our proposed machine learning based architecture. We conclude in general that resampling techniques for resolving data imbalance problem in extractive meeting summarization domain, results in enhanced system performance.

We are currently working on multiple extensions of this work, including investigating how the results can be applied to other corpora, adding additional features, and finally methods for post-processing extractive summaries.

Acknowledgments This work is supported by the European IST Programme Project AMIDA [FP6-0033812]. This paper only reflects the authors views and funding agencies are not liable for any use that may be made of the information contained herein.

References

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. In *Educational and Psychological Measurement*.

Peter Corbett and Ann Copestake. 2008. Cascaded classifiers for confidence-based chemical named entity recognition. In *Current Trends in Biomedical Natural Language Processing*.

H. P. Edmundson. 1969. New methods in automatic extracting. In *J. ACM, 16(2)*.

Erhard W. Hinrichs, Katja Filippova, and HolgerWunsch. 2005. A data-driven approach to pronominal anaphora resolution for german. In *In Proceedings of Recent Advances in Natural Language Processing*.

Gabriel Murray. 2008. *Using Speech-Specific Characteristics for Automatic Speech Summarization*. Ph.D. thesis, University of Edinburgh.

Shasha Xie, Yang Liu, and Hui Lin. 2008. Evaluating the effectiveness of features and sampling in extractive meeting summarization. In *IEEE Spoken Language Technology Workshop (SLT)*, pages 157–160.

Xiaodan Zhu and Gerald Penn. 2006. Summarization of spontaneous conversations. In *Proceedings of the 2006 ACM Conference on Computer Supported Cooperative Work (CSCW 2006),*.