Comparative Study on Subject Classification of Academic Videos using Noisy Transcripts

Hau-Wen Chang Hung-sik Kim Shuyang Li[+] Jeongkyu Lee[+] Dongwon Lee

The Pennsylvania State University, USA [+]University of Bridgeport, USA

{hauwen,hungsik,dongwon}@psu.edu, [+]{shuyangl,jelee}@bridgeport.edu

***Abstract*—With the advance of Web technologies, the number of "academic" videos available on the Web (e.g., online lectures, web seminars, conference presentations, or tutorial videos) has increased explosively. A fundamental task of managing such videos is to classify them into relevant subjects. For this task, most of current content providers rely on keywords to perform the classification, while active techniques for automatic video classification focus on utilizing multi-modal features. However, in our settings, we argue that both approaches are not sufficient to solve the problem effectively. Keywords based method is very limited in terms of accuracy, while features based one lacks semantics to represent academic subjects. Toward this problem, in this paper, we propose to transform the video subject classification problem into the text categorization problem by exploiting the extracted transcripts of videos. Using both real and synthesized data, (1) we extensively study the validity of the proposed idea, (2) we analyze the performance of different text categorization methods, and (3) we study the impact of various factors of transcripts such as quality and length towards academic video classification problem.**

I. Introduction

With the rapid development of technologies in software and hardware, users can now access a large number of videos on the Web. The trend makes an impact on the way scholastic findings is disseminated, which is conventionally through research articles in publication outlets. Scholars now may summarize their findings in a short video clip and circulate it for wider dissemination in online video community (e.g., YouTube) or educational video web sites (e.g., TeacherTube and ScienceStage). Similarly, increasingly more number of publishing venues record author's presentation as a video clip and share them on the Web (e.g., VideoLectures). Furthermore, many schools start to post instructors' lectures as video clips to the Web for wider audience (e.g., Open Yale Courses). Potentially, such multimedia-rich media, which we refer to as **academic videos** in this paper, are very useful for many people. While there have been many researches on video retrieval and repository systems, few exists that specifically targets at only the academic videos. In the **Leedeo** project [1], we aim at building such a large-scale video search engine that crawls, gathers, indexes, and searches "academic" videos from the Web.

When such a large number of academic videos are gathered from the Web, automatically classifying those videos by subjects is one of the fundamental functions. With a good quality subject classification, such academic videos can be browsed and searched more effectively and efficiently. However, often, those academic videos downloaded from the Web do not come with appropriate genre labels, or some videos are tagged only with very specialized terms. Therefore, it is not easy to come up with globally agreeing genre labels, when multiple videos are considered together. To cope with these issues, an automatic method for classifying academic videos is desirable.

While extensive research for video classification exists in literature, by and large, they extract distinct *audio* (i.e., zero-crossing rate (ZCR) [2] and discrete cosine transform (DCT) [3]), *visual* (i.e., colors [4] and motions [5]), or *textual* features (e.g., OCR [6] and closed captions [7]) from videos, and use them for classification. However, the challenge that we face in the **Leedeo** project is that academic videos seldom have such distinct audio, visual, or textual features. For instance, in the majority of academic videos that we have collected, visual features are monotonous such that one or two speakers speak in front of a board or screen plainly without much characteristic movement. Therefore, it is hard to use the visual features to differentiate a conference presentation from a college lecture video. Similarly, dominant audio features such as gun shot or song playing do not exist in academic videos.

Text-based approach for video classification uses the metadata of videos such as title, speaker's profile, tagged data or abstract associated with the videos. However, when academic videos are crawled and gathered from the Web in the **Leedeo** project, often, identifying and extracting such metadata of videos is a non-trivial and erroneous task. That is because many of videos do not have accurate metadata to use in classification. Due to this unique idiosyncrasy of academic videos, most of conventional video classification techniques (e.g., [2], [3], [4], [5]) are not readily applicable for academic videos.

Toward this unique challenge, therefore, what we propose in this paper is to exploit the extracted transcripts of academic videos for subject classification. Since speakers must speak verbally, there exists almost always an audio channel embedded in each academic video. By transforming this audio channel into transcripts using either human experts or automatic speech recognition (ASR) software, each video can be associated with a rich corpus that describes the contents of the video well. Then, by applying conventional text categorization

techniques developed in IR community to these extracted transcripts (as documents), we expect to improve the accuracy of video classification substantially. Toward this proposal, in this paper, as the first step, we extensively study the validity of the proposed idea and compare the impacts of various transcripts with respect to their length and quality toward the accuracy of academic video classification.

II. Related Work

A. Automatic Video Classification

Automatic video classification has been an active research area in recent year, which classifies a given video into one of the predefined categories. The categories usually have conceptual or semantic meanings, such as genres, subjects or topics. For example, [8] classifies movies by its genre, i.e., comedy, horror, action and drama/other movies, while [7] categorizes news videos into topics including politics, daily events, sports, weather, entertainment, etc.

The classification methods utilize one of more features (i.e., multi-modal or fusion-based approach) extracted from a video, such as *audio* features [2], [3], *visual* features [4], [5] and *textual* features [6], [7]. However, several problems will be faced when exploiting a multi-modal approach in academic video classification. For the audio-visual features, most of academic videos have only speeches without dominant sound, so it would be difficult to characterize any academic subjects using the sound itself. Moreover, academic videos usually have a small number of moving objects with little movement. The range of the camera motion and shot changes are also very limited. These monotonic visual features do not fit to our classification task. The textual features seem the only feasible approach, so we are motivated to extract features from transcript for our classification task.

B. Spoken Document Retrieval and Classification

One notable earlier contribution on searching spoken document is the Spoken Document Retrieval (SDR) track in TREC 6 to TREC 9[9]. The goal is to investigate the retrieval methodologies on corrupted documents generated by ASR software.On this evaluation-driven task, the participant implements a SDR system including two components, i.e., ASR and IR. There are three tasks for the IR component: (i) reference retrieval where a (nearly) perfect transcript is used, (ii) baseline retrieval where a ASR-generated prepared by NIST is used, and (iii) speech retrieval where ASR-generated transcripts participant's ASR component is used. A conclusion made from the result is that retrieval performance and word error rates have a near-linear relationship; however, the performance degrades very gently (5%) for increasing recognition errors (10% $\sim$ 35%). With the fact that the speech retrieval has a similar performance to reference retrieval with a large collection spoken documents, SDR is claimed as a solved problem.

III. Video Classification as Text Categorization Problem

Since the core intuition of our proposal is to treat the extracted transcripts of academic videos as documents and to apply conventional text categorization techniques to these documents, in this section, we cover three popular text categorization methods that will be compared in our experiments (see Section IV).

The task of *text categorization* [10], [11] assigns a Boolean value to each pair $\langle d_j, c_i \rangle$ belonging to $D \times C$, where D is the domain of documents and $C = \{c_1, c_2, \ldots, c_N\}$ is the set of predefined categories. A value of T (true) assigned to $\langle d_j, c_i \rangle$ indicates that a document, d_j, is under a category, c_i, while a value of F (false) indicates otherwise. In other words, the text categorization task is to approximate the unknown target function $R' : D \times C \rightarrow \{T, F\}$, by means of a function $R : D \times C \rightarrow \{T, F\}$ called the *classifier* such that R' and R coincide as close as possible.

In this paper, we exploit three well known classifier, i.e., Naive Bayes, KNN, and SVM, based on supervised learning approach. In other words, with labeled documents (i.e., the belongings of the categories are known), the proper classifier is acquired using learning methods or algorithms by obtaining the classification function of R. These methods are well-studied and known as good text classifiers in the literature. However, the performance has not been verified as *spoken* document classifiers, which, as transcripts extracted from academic videos in our setting, are almost always very *noisy* due to several reasons: (1) Since academic videos are more domain specific than regular videos (e.g., news or sports videos) are, often, vocabularies in academic videos are more peculiar and technical. Therefore, the accuracy of ASR software drops significantly for academic videos; and (2) Since the majority of academic videos are still produced in non-professional environment, the quality of audio in them is much poorer than are the videos professionally produced one, i.e., broadcasting companies. Such poor quality of transcript causes the accuracy of ASR software significantly to degrade when it is applied to academic videos.

A. Naive Bayes

A naive Bayes classifier is a simple probabilistic classifier based on Bayesian theorem and is especially appropriate when the dimension of feature space is high [11]. Despite its simplicity, Naive Bayes can often outperform sophisticated classification methods. The probability of a document d being in class c is computed as:

$$P(c|d) \propto P(c) \prod_{1 \leq k \leq n_d} P(t_k|c) \tag{1}$$

where $P(t_k|c)$ is the conditional probability of token t_k occurring in a document of class c and n_d is the total number of tokens. To find the most probable class in text categorization,

maximum a posteriori (MAP) class c_{map} is computed :

$$c_{map} = \arg\max_{c \in C} \hat{P}(c|d) = \arg\max_{c \in C} \hat{P}(c) \prod_{1 \le k \le n_d} \hat{P}(t_k|c) \quad (2)$$

B. KNN

K-nearest neighbor (KNN) is one of the simplest machine learning algorithms [12]. A document is classified by the majority of K closest neighbors. For example, if $K = 1$, the unknown document will be classified by the known nearest neighbor (or class). KNN consists of two steps like other supervised algorithms. In the training step, the KNN classifier selects training set of documents and sprays them to the appropriate positions of the multi-dimensional field for the proper uses in the classification step. The unknown test documents are classified by K nearest known neighbors positioned in the training step. We use Euclidean distance in our feature space, i.e., $distance_i = ||d_i - d_0||$, where d_0 is a test document and d_i is a training document.

C. SVM

The support vector machine (SVM) classifier as one of the state-of-the-arts learning methods finds boundaries in input feature space [13]. In its simplest form, i.e., binary classification, SVM finds a boundary with maximum margin between the two classes in the feature space. If the data are not linearly-separable, the margin may be charged with penalty and the boundary may be found in a higher or infinite dimensional space.

IV. Empirical Validation

A. Set-Up

The Naive Bayes (NB) and KNN algorithms are implemented in Matlab 7.0, while SVMlight [14] is used for the SVM implementation. All experiments were conducted on a HP Desktop with Intel Quad-core 2.4GHz, 4G RAM, and Windows Vista OS.

Data Set. To experiment our proposal, ideally, we need a data set that have sufficient number of video clips with various subjects. Moreover, to understand the impact of noise in transcripts, the data set should have human-transcribed *perfect-quality*[1] transcripts. As a raw data set that meets these requirements, we chose the `Open Yale Courses` project[2] that provides a collection of introductory course videos and transcripts taught by faculty at Yale University. The collection includes 25 courses from 17 different departments whose names are used as the subjects, e.g. physics, psychology, etc. The number of lectures in each subject ranges from 20 to 37 while the length of lectures in each subject ranges from 35 to 90 minutes. The total running time is over 585 hours. The Yale data set provides different formats of media for browsing such as video, audio, and human-transcribed transcripts.

Speech Recognition. The work of our speech recognition is based on Sphinx 4.0, one of the state-of-the-art hidden Markov model (HMM) speech recognition systems which is a open source project from CMU [15]. In this pluggable framework of speech recognition, three main components exist, i.e., FrontEnd, Linguist, and Decoder. In our experiment, we use WordPruningBreadthFirstSearchManager as the search manager in the decoder and LexTreeLinguist as the Linguist. We adopt HUB4 acoustic model that have been trained using 140 hours of 1996 and 1997 hub4 training data which are continuous speeches from broadcast news [16]. We also use HUB4 language model which is a trigram model built for tasks similar to broadcast news. The vocabulary includes 64000 words.

Synthesized Test Sets. From the original Yale data set, next, we synthesize multiple test data sets for our experiments as follows:

(1) *Cross Validation*: To divide the data set into training and testing sets, two splits are considered: 5-fold *cross validation (CV)* and *course split (CS)*. First, in the CV test set, for each subject, randomly chosen 80% of transcripts are used for training, while remaining 20% for testing. This CV process repeats 5 times, and we measure the average at the end. Second, in the CS test set, for those subjects with more than 1 course (e.g., `English` and `History`), one course is chosen as the training set while another as the testing set. While CV test sets aim to test whether subject can be determined given part of the lectures of a course, CS test sets aim to test whether subject can be determined given a complete new course that has not been studied for a classifier. Therefore, in general, CS test sets are more difficult than CV test sets.

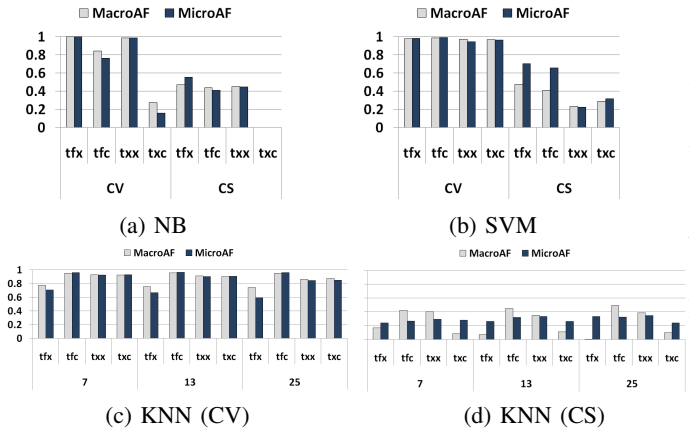
(2) *Term Weighting*: Text categorization methods take a term matrix as an input, which is extracted from transcripts with various term weighting schemes. There are 4 term weighting schemes used in our experiments, i.e., `txx`, `tfx`, `txc` and `tfc` (using the notations from SMART system [17]), where `t`, `f`, `c`, and `x` indicates raw term frequency, inverse document frequency, cosine normalization, and none, respectively. For each transcript, we prepare 4 versions with `txx`, `tfx`, `txc` and `tfc` weighting schemes. The size of the overall term matrix is 31,762 terms and 634 documents after Porter's stemming.

(3) *Quality of Transcripts*: To compare the impact of the quality of transcripts in video classification, we compare three test sets: (i) a perfect transcript by human scribers (PF), (ii) an erroneous transcript with synthetic errors by Edit operations (i.e., insert, delete, and substitute) (SE), and (iii) an erroneous transcript by ASR software (SR). The quality of transcript is estimated by *Word Error Rate (WER)*, which is based on minimum edit distance of the erroneous transcript relative to a perfect one [18] defined as: $\text{WER} = \frac{\text{\# of insert+delete+substitute}}{\text{\# of words in the perfect transcript}}$.

(4) *Length of Transcripts*: Finally, full version *vs.* abbreviated test sets of transcripts are examined to see the impact of

[1] While it is possible for human-transcribed transcripts may still have errors, compared to automatically generated ones, we believe their quality must be much more superior. Therefore, in this experiment, we consider human-transcribed transcripts as "perfect" ones.

[2] http://oyc.yale.edu/



(a) NB (b) SVM

(c) KNN (CV) (d) KNN (CS)

Fig. 1. Precisions using perfect transcripts.

the length of transcripts for classification. For the abbreviated versions, we use the simple scheme of the first 10% of the first 10 minutes of total length of each transcript.

Evaluation Metrics. For the evaluation metrics, we consider F-measure in two popular averaging metrics to calculate the result of our multi-label classification task, i.e., macro-averaged F-measure (MacroAF), and micro-averaged F-measure (MicroAF). The macro/micro-averaged F-measures are determined by the macro/micro-averaged precision and recall. To calculate macro-averaged precision (MacroAP), the precision for each class is firstly calculated separately and then the MacroAP is calculated by taking the average of precisions across all classes. On the other hand, in micro-averaged precision (MicroAP), each transcript is given an equal weight so the MicroAP is the average precision of the data set as a whole. Formally, the averaging metrics are defined as:

$$\text{MacroAP} = \frac{1}{|K|}\sum_{i=1}^{|K|}\frac{TP_i}{TP_i + FP_i}, \text{MicroAP} = \frac{\sum_{i=1}^{|K|} TP_i}{\sum_{i=1}^{|K|} TP_i + FP_i}$$

, where TP_i and FP_i are the number of true positive and false positive for class i, respectively. MacroAR and MicroAR are defined in a similar fashion. Macro-average F-measure (MacroAF) and micro-average F-measure (MicroAF) are defined as the harmonic mean of MacroAP and MacroAR, and that of MicroAP and MicroAR, respectively. The macro/micro-averaged recalls (MacroAR/MicroAR) are defined in a similar fashion. Finally, MacroAF and MicroAF are defined as the harmonic mean of MacroAP and MacroAR, and that of MicroAP and MicroAR, respectively.

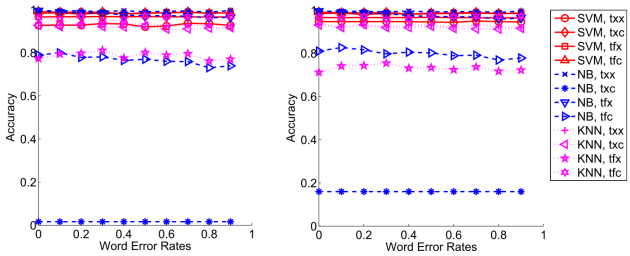
B. Result of Perfect Transcript

First, we assess the result of the baseline test, i.e., subject classification using *perfect* transcripts. Figure 1 (a) shows the baseline result of Naive Bayes (NB) method. For CV test set, the `tfx` weight scheme has the highest MacroAF and MicroAF (i.e., around 0.99), while the `txc` has the lowest MacroAF and MicroAF (i.e., 0.06 and 0.16, respectively). For much harder CS test set, overall F-measure drops by 40% from CV case. Among weighting schemes, it appears that `t` (term frequency) and `f` (inverse document frequency) play the major role, while somehow `c` (cosine normalization) works against. Note that in the CS test set, if two courses A and B are significantly different in their contents even if both belong to the same subject (say two courses "Computer Vision" and "Relational Databases" in Computer Science), then training using one course does not necessarily help in testing the other course. Therefore, in general, we expect a sharp drop in precisions when we compare CV to CS case. Second, the baseline result for the SVM method (with linear kernel and $C = 1$) is shown in Figure 1(b). All 4 term schemes have good F-measure ranging from 0.92 to 0.99. The F-measure for `tfx` and `tfc` against CS test set are around 0.6, while those of `txx` and `txc` are below 0.3. Overall, for CV test set, SVM method shows a promising result, especially compared to the other two text categorization methods. Finally, Figures 1(c)–(d) show the baseline result of the KNN method with K = 7, 13, and 25. All term schemes have F-measures exceeding 0.9, except for `tfx` weight scheme whose precision is around 0.6% $\sim$ 0.7%. For a much harder CS test set, F-measures rapidly drop compared to CV case. The best weighting schemes for CS case are `tfc` and `txx` which achieve F-measures around 0.4. On this baseline test, we observed that NB and KNN are more subject to the choice of weight scheme, and that micro-averaging and macro-averaging have similar results on our dataset.

C. Effect of the Noise of Transcript

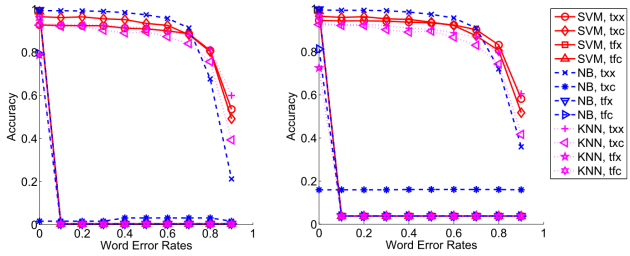
Although there are a large number of academic videos available on the Web, the availability of their perfect transcripts are very limited since human transcription is an expensive and time-consuming task. Meanwhile, automatic speech recognition (ASR) software can generate imperfect machine-transcribed transcripts at decent speed and cost. At the time of the writing, however, the word error rate (WER) of the state-of-the-art ASR system is around 10% for broadcast news and around 20% for conversational telephone speech as reported in [18]. Moreover, as discussed in Section III, when ASR system is applied to academic videos, its WER increases even further due to poor audio quality and peculiar vocabularies used. While conventional text categorization methods perform relatively well (especially for CV test sets) in Section IV-B, we wonder if that remains true with "noisy" transcripts which are more commonly encountered in practical settings. However, an ASR recognizer interacts with several models and the complexity limits the possibility of predicting its behavior. Instead of building a model of possible output from an ASR recognizer, we take a approach of simulation to study relation of different levels of noisy transcripts and classification performance in term of precision.

Synthetically Erroneous Transcripts (SE). The synthetically erroneous transcript allows us to simulate different levels of WERs and the impact of edit operations therein. In our experiment, 3 different edit operations (i.e., `add`, `delete`, and `substitute`) are simulated independently on all the perfect transcripts as follows: (1) *Addition*: a number of terms for specific error rate are selected in a uniformly random fashion



(a) MacroAF (b) MicroAF

Fig. 2. Precision of SE test set with addition.



(a) MacroAF (b) MicroAF

Fig. 3. Precision of SE test set with deletion.

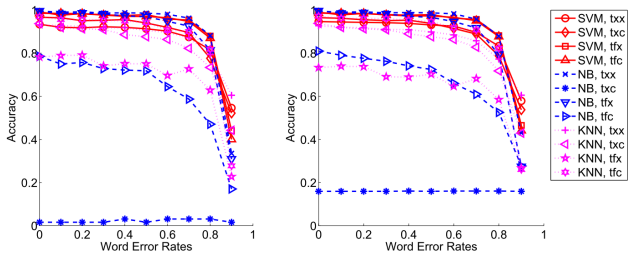
and their term frequency is added by one. A term can only be selected once; (2) *Deletion*: a fixed number (100 by default) of terms whose frequency are greater than zero are selected in a uniformly random fashion, and their term frequency is deducted by one accordingly. This procedure repeats until the total number of desirable deletions is achieved; and (3) *Substitution*: we performs aforementioned deletion followed by the addition.

The MacroAF and MicroAF for addition are shown in Figure 2. The SVM method performs well and remains stable for all 4 term weighting schemes. The NB has the best precision with `txx` and `tfx` but the performance with `tfc` slightly decreases when WER goes high. Again, `txc` does not work well in this task. Precision for KNN (K=7) ranges from 0.7 $\sim$ 0.9 except with `tfc` which is around 0.8. In conclusion, when transcripts have errors due to addition operations, the performance changes depending on the chosen weighting schemes, but it remains robust even if WER increases. Both MacroAF and MicroAF have very similar pattern.

Figure 3 shows the result with deletion. For SVM, only `txx` and `txc` work well initially but their precisions decrease as WER increases, especially WER exceeds 0.6. The results of KNN are similar to those of SVM but a little worse for all 4 weighting schemes. NB has only `txx` working, whose precision drops sharply after WER of 0.7. Figure 4 shows the result for substitution. When WER is low, the results are similar to the results of insertion for 4 weighting schemes. The performance slightly decreases as WER increases and drops sharply after WER of 0.8.

D. Effect of the Length of Transcripts

In this section, we investigate the impact of length of the transcript to the video classification. We test 3 cases on the length of transcript including the first 1%, 5%, and 10% of



(a) MacroAF (b) MicroAF

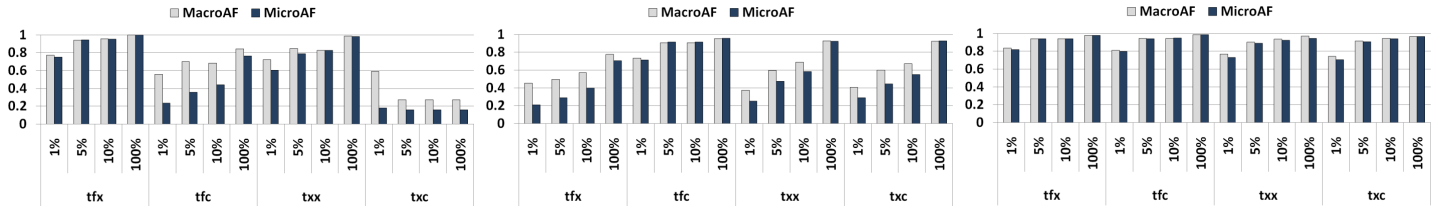
Fig. 4. Precision of SE test set with random substitution.

perfect transcript and ASR-generated transcript. The intuition to shorten the length of transcripts by taking the first 1–10% is that in many academic videos such as lecture or presentation videos, speakers often give a summary of the talk in the beginning. On average, the length of the first 10% of perfect transcripts and the first 10 minutes of ASR-generated transcripts is around 600 and 1000 words, respectively.

Figure 5 shows a comparison of all three methods on different length of perfect transcripts with CV grouped by weighting schemes. In Figure 5(a), `tfx` scheme shows the best accuracy, i.e., using only the first 1% of transcripts, NB could achieve the F-measure of 0.9. In Figure 5(b), `tfc` scheme which has the best performance, shows roughly the same results using the 100%, 10%, and 5% of transcripts, but has a sharp drop from 5% to 1%. The F-measure of other schemes also decrease, but have larger differences when moving from 100% to 10% and to 5%. In Figure 5(c), SVM has high F-measure regardless of the chosen weighting scheme. This is similar to the results of `tfc` with KNN and `tfx` with NB. We believe if a method (with a specific weighting scheme) has a strong precision and recall, the length of transcripts used in the classification has less impact on them. Even with, 1% of the length, it is still possible to achieve the F-measure of 0.8. On the other hand, above a threshold of certain length, the F-measure becomes stable.

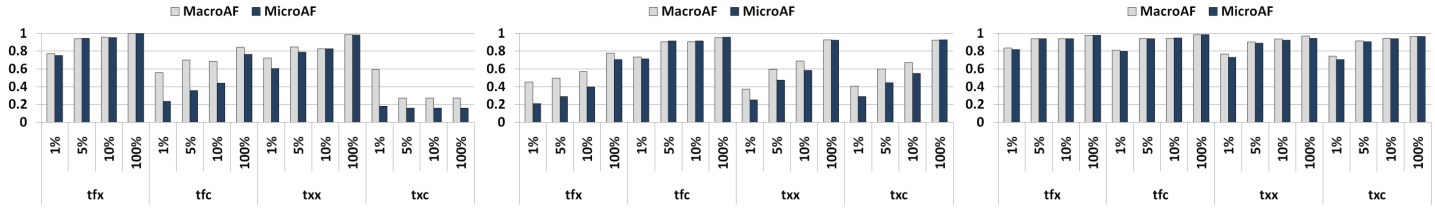
Figure 6 shows the result of using the different length of ASR-generated transcript. For NB, `tfx` shows the best result in which MicroAF reaches 0.81 with the first 5% transcripts. MicroAF reaches 0.92 with the first 10% transcript, which is comparable to the result on the first 10% perfect one, 0.95. For KNN, only `tfc` can be considered effective, whose F-measure is 0.5 with the first 1% of transcript, and improves to 0.80 with the first 10%. For SVM, all weighting schemes show similar performance. The F-measures are between 0.5 $\sim$ 0.6 on the first 1% transcript, and improve to 0.8 $\sim$ 0.9 on the first 5% one.

From this experiment, we observed that, with a good choice of term weighting, the result of the first 5% of full length is comparable to that of full length transcript. For ASR-generated transcript, we need at least the 10% to reach similar performance. Moreover, we may conclude that for: (1) The text categorization is susceptible to the choice of term weighting scheme; (2) The CS test set is more difficult to classify and may need be studied further; and (3) The impact of the quality and length of transcript exists, but not substantial. Validated



(a) Naive Bayes (NB) (b) KNN (K=7) (c) SVM

Fig. 5. Comparison using different length of perfect transcripts on CV.



(a) Naive Bayes (NB) (b) KNN (K=7) (c) SVM

Fig. 6. Comparison using different length of ASR-generated transcripts on CV.

with the CV test set, all three text categorization methods can achieve the F-measure over 0.8 with a good choice of weight scheme. However, in general, SVM shows the most robust performance.

V. Conclusion and Future Work

We have conducted comparative study on the subjects classification of academic videos with various methods. For the study, we transformed the video subject classification problem into the text categorization one by exploiting the transcripts of videos. We also investigated the impact of plausible factors, such as noise/quality of transcripts and the length of a video with three popular text categorization methods. Our experimental results shows that SVM promises the best result in both CV and CS cases. In terms of *term weights*, `tfx` is good in NB and SVM, but not in KNN, while `tfc` is good in SVM and KNN, but not in NB. From the synthetically erroneous transcripts test, we observed that the learning methods with a good choice of weighting scheme is very robust even though 70% of the transcript is incorrect. In addition, the learning methods is dependable when only a small part of transcript, human or machine transcribed, are available.

Our future work is directed towards the followings:(1) With the promising results from the experiment, we plan to apply the subject classification methods to academic video search engine that we are currently developing; (2) The performance of the learning methods on CS test set needs more improvement. We will seek a solution to improve the performance of this realistic setting; and (3) The evaluation on the synthetically erroneous transcripts may not truly reflect the errors occurred in a real transcript. Moreover, for transcripts with the same WER, the performance on classifying them may be different. We plan to investigate more accurate WER model for the simulation.

References

[1] D. Lee, H. S. Kim, E. K. Kim, S. Yan, J. Chen, and J. Lee, "Leedeo: Web-crawled academic video search engine," in *ISM*. IEEE Computer Society, 2008, pp. 497–502.

[2] E. Wold, T. Blum, D. Keislar, and J. Wheaten, "Content-based classification, search, and retrieval of audio," *Multimedia, IEEE*, vol. 3, no. 3, pp. 27–36, Fall 1996.

[3] B. Logan, "Mel frequency cepstral coefficients for music modelingl," in *1st International Symposium on Music Information Retrieval (ISMIR)*, October 2000.

[4] A. Gupta and R. Jain, "Visual information retrieval," *Communication of the ACM*, vol. 40, no. 5, pp. 70–79, 1997.

[5] G. Iyengar and A. Lippman, "Models for automatic classification of video sequences," in *SPIE*, 1998.

[6] A. G. Hauptmann, R. Yan, Y. Qi, R. Jin, M. G. Christel, M. Derthick, M. Chen, R. V. Baron, W. Lin, and T. D. Ng, "Video classification and retrieval with the informedia digital video library system," in *TREC*, 2002.

[7] W. Zhu, C. Toklu, and S. Liou, "Automatic news video segmentation and categorization based on closed-captioned text," in *IEEE ICME*, Aug. 2001, pp. 829–832.

[8] Z. Rasheed and M. Shah, "Movie genre classification by exploiting audio-visual features of previews," in *IEEE ICPR*, Quebec City, Canada, 2002, pp. 1086–1089.

[9] J. S. Garofolo, C. G. P. Auzanne, and E. M. Voorhees, "The trec spoken document retrieval track: A success story," in *in Text Retrieval Conference (TREC) 8*, 2000, pp. 16–19.

[10] F. Sebastiani and C. N. D. Ricerche, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, pp. 1–47, 2002.

[11] C. D. Manning, P. Raghavan, and H. Schutze, *Introduction to Information Retrieval*. Cambridge Univ. Press, 2008.

[12] S. Tan, "An effective refinement strategy for KNN text classifier," *Expert Syst. and Appl.*, vol. 30, no. 2, pp. 290–298, 2006.

[13] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2001.

[14] T. Joachims, "Making large-scale SVM learning practical," *Advances in Kernel Methods–Supp. Vector Learning*, 1999.

[15] W. Walker, P. Lamere, P. Kwok, B. Raj, R. Singh, E. Gouvea, P. Wolf, and J. Woelfel, "Sphinx-4: a flexible open source framework for speech recognition," Mountain View, CA, USA, Tech. Rep., 2004.

[16] P. Placeway, S. Chen, M. Eskenazi, U. Jain, V. Parikh, B. Raj, M. Ravishankar, R. Rosenfeld, K. Seymore, M. Siegler, R. Stern, and E. Thayer, "The 1996 hub-4 sphinx-3 system," in *In Proc. of DARPA Speech Recognition Workshop*. The, 1996.

[17] G. Salton, *The SMART Retrieval System—Experiments in Automatic Document Proc.* NJ, USA: Prentice-Hall, 1971.

[18] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognitiong*. Prentice-Hall, 2009.