Improving the Transcription of Academic Lectures for Information Retrieval

Audrey Mbogho
Department of Computer Science
University of Cape Town
Rondebosch, South Africa
Email: Audrey.Mbogho@uct.ac.za

Stephen Marquard
Centre for Educational Technology
University of Cape Town
Rondebosch, South Africa
Email: Stephen.Marquard@uct.ac.za

***Abstract*—Recording university lectures through lecture capture systems is increasingly common, generating large amounts of audio and video data. Transcribing recordings greatly enhances their usefulness by making them easy to search. However, the number of recordings accumulates rapidly, rendering manual transcription impractical. Automatic transcription, on the other hand, suffers from low levels of accuracy, partly due to the special language of academic disciplines, which standard language models do not cover. This paper looks into the use of Wikipedia to dynamically adapt language models for scholarly speech. We propose Ranked Word Correct Rate as a new metric better aligned with the goals of improving transcript searchability and specialist word recognition. The study shows that, while overall transcription accuracy may remain low, targeted language modelling can substantially improve searchability, an important goal in its own right.**

I. Introduction

Lecture capture technologies are gaining popularity in higher education. However, a single continuous recording is often unhelpful for users. As students often use lecture recordings for revision or preparation for assessments [1], they may wish to play back a part rather than the whole of a lecture, or identify lectures containing particular material. To support this, various indexing schemes have been used to enable faster navigation and searching. For example, slide images are commonly used to provide a visual index within the lecture. However, a transcript of the lecture provides even more possibilities, as it enables quick navigation within the lecture, discovery through text search across lectures within the lecture capture system, and discovery through search engines and content aggregators.

In many contexts, producing manual transcripts from audio recordings is not economically viable as it is time-consuming and expensive. Using automated speech recognition (ASR) technologies for transcription is thus an attractive lower-cost approach. At the same time, ASR systems are imperfect and may introduce many errors into a transcription. Key factors affecting accuracy include:

1) the audio quality of the recording, influenced by the type of microphone used, venue acoustics and amount of background noise
2) whether the recognition system has been trained for a particular speaker, or is using a speaker-independent acoustic model
3) for speaker-independent systems, the match between the speaker's accent and the acoustic model
4) the match between the vocabulary and pronunciation in the lecture with the language model and pronunciation dictionary.

Due to the above factors, many transcripts fall short of the accuracy threshold for readability [2]. Although these transcripts are unusable as a substitute for the recording itself, they can still be useful for search and navigation. In this study, we focused on the fourth factor above, with the aim of enhancing the linguistic match between the lecture and the ASR system's language resources. A good match then enables information retrieval activities: (1) identifying lectures in which search terms occur, and (2) identifying the points within a lecture where search terms occur. The motivation here is that specialist words are likely search terms, and their recognition is therefore critical to information retrieval in the academic domain.

For most ASR systems, vocabulary represents a "hard" constraint. While other factors such as audio noise or accent mismatch may be present to a greater or lesser degree and influence the transcription accuracy accordingly, if a word is not in the dictionary and language model, it will never be recognised.

II. Related Work

There is much ongoing work to address the four key challenges mentioned above. Here, however, we only review work that addresses the language mismatch challenge as this is the focus of our own research. In addition we examine the metrics used to decide how good an automatically generated transcript is, a discussion which serves to motivate our proposal of a new metric.

A. Language model adaptation

Researchers have investigated strategies for generating and adapting the language model (LM) to improve recognition accuracy for lectures, on the assumption that a model which closely reflects the context of the utterances is likely to outperform a more generic language model. Kato *et al.* investigated the use of a topic-independent LM derived from a large corpus of text from lecture transcripts and panel discussions, with topic-specific keywords removed [3]. The model is then

adapted to specific lectures by using the preprint paper of the lecture to be delivered (when available).

The use of lecture slides for adapting the LM has been explored by several research groups. Yamazaki *et al.* note that a "a strong correlation can be observed between slides and speech" and explore first adapting the LM with all text found in the slides, then dynamically adapting the LM for the speech corresponding to a particular slide [4].

Munteanu *et al.* pursue an unsupervised approach using keywords found in slides as query terms for a web search. The documents found in the search are then used to adapt the LM [5].

Kawahara *et al.* investigate three approaches to adapting the LM, viz. global topic adaptation using Probabilistic Latent Semantic Analysis (PLSA), adaptation with web text derived from keyword queries and dynamic local slide-by-slide adaptation using a contextual cache model. They conclude that the PLSA and cache models are robust and effective, and give better accuracy than web text collection because of a better orientation to topic words [6]. Latent Semantic Analysis is an approach to document comparison and retrieval which relies on a numeric analysis of word frequency and proximity.

B. Measuring the accuracy of lecture transcripts

The most widely used accuracy metric for recognition tasks is the Word Error Rate (WER), computed as the Levenshtein distance ("edit distance") between the recognized text and a reference transcript. This is the number of insertions, deletions and substitutions required for the hypothesis to match the reference transcript, as a proportion of the number of words in the reference transcript. A related measure is the Word Correct Rate (WCR), which ignores insertion errors.

While its widespread use makes WER a useful measure to compare competing approaches, it may often not account for the actual impact of errors for the application at hand. For example some errors may be more trivial than others and easily overlooked, while keyword accuracy may be disproportionately significant.

Bourlard *et al.* have taken issue with WER's dominance in the field, arguing that reliance on reporting WER may in fact be counter-productive, undermining the development of innovative new approaches and "deviant" research paradigms [7].

McCowan *et al.* point out that WER characterises recognition performance as a string editing task, whereas for many applications speech recognition is better understood as supporting information retrieval tasks [8]. Cited weaknesses of WER include that it is not a proper rate (as it can range below 0 and above 1), is not easily interpretable and cannot be decomposed in a modular way.

Park *et al.* examine automatic transcripts from the perspective of information retrieval (IR), investigating the effects of different recognition adaptations on WER and the IR measures precision and recall, which relate to matches of keyword query strings in the recognized text [9]. Results show that good retrieval performance is possible even with high error rates, and conversely that adapting the language model with spontaneous speech data improves accuracy, but is of marginal value to information retrieval tasks. Wang *et al.* argue against WER, reporting results where an alternate language model produced higher word error rates but better performance with an alternate task-oriented metric, slot understanding error [10].

McCowan *et al.* propose four qualities for an improved metric: that it should be a direct measure of ASR performance, calculated in an objective, automated manner, clearly interpretable in relation to application performance and usability, and modular to allow application-dependent analysis [8].

To better characterise searchability in terms of keyword recognition, we introduce a new metric, Ranked Word Correct Rate (RWCR), which is a modified version of WCR. While WCR takes into account all recognised words, RWCR calculates the total recognition rate of those words in the transcript which occur *below a given frequency rank* in general English. Thus the recognition accuracy of unusual words (e.g. Comus) affects the recognition score, while the recognition accuracy of common words (e.g. a, the, and) is ignored.

C. Experimental Setup and Data Preparation

Experimental setup involved the following choices.

Linguistic Resource Wikipedia was selected for its broad coverage of language, including academic language.

ASR Engine CMU Sphinx is an open source speech recognition toolkit from Carnegie Mellon University. The version of Sphinx selected for this project is Sphinx4, a highly customizable recognition engine written in Java.

Language Model The HUB4 language model was used for establishing baseline recognition performance. It is referred to herein as the reference model.

Audio data 13 lectures from Open Yale Courses (OYC) were selected for audio quality, subject variety, and availability of transcripts. The course names and lecture titles are listed below. (The code in brackets next to each course will be used in later sections to refer to the course.)

1) Frontiers and Controversies in Astrophysics (astr160) – Dark Energy and the Accelerating Universe and the Big Rip
2) Frontiers of Biomedical Engineering (beng100a) – Cell Culture Engineering
3) Frontiers of Biomedical Engineering (ben100b) – Biomolecular Engineering: Engineering of Immunity
4) Principles of Evolution, Ecology and Behavior (eeb122) – Mating Systems and Parental Care
5) Milton (engl220) – Lycidas
6) The American Novel Since 1945 (engl291) – Thomas Pynchon, The Crying of Lot 49
7) Introduction to Theory of Literature (engl300) – The Postmodern Psyches
8) The American Revolution (hist116) – The Logic of Resistance
9) European Civilization, 1648-1945 (hist202) – Maximilien Robespierre and the French Revolution
10) Death (phil176) – Personal identity, Part IV; What matters?

11) Introduction to Political Philosophy (plsc114) – Socratic Citizenship: Plato, Apology
12) Introduction to Psychology (psyc110) – What Is It Like to Be a Baby: The Development of Thought
13) Introduction to New Testament History and Literature (rlst152) – The "Afterlife" of the New Testament and Postmodern Interpretation

D. Wikipedia as a linguistic resource

The English Wikipedia (hereafter Wikipedia) is used to create three types of resource for this project:

1) a dictionary of English words with word frequency counts
2) a generic language model, approximating general English usage
3) topic-specific language models, approximating English usage in a topic area

Advantages of using Wikipedia for this purpose include:

- It is a large corpus, containing more than 4 million articles and over 1000 million words. It is thus of a similar order of magnitude to resources such as the English Gigaword Corpus [11].
- It has been shown to be a usable language resource for other natural language processing tasks [12].
- Wikipedia articles include semantic metadata through inter-article links and other tags such as categories. This semantic structure can be used to select subsets of Wikipedia articles.
- It has broad topic coverage.
- It is updated continuously, and thus dynamic and contemporary.
- Wikipedia text is available at no cost, and published with a permissive license allowing derivative works to be freely redistributed [13].

The principle disadvantage is that it is a loosely curated resource, and thus contains a greater number of typographical, spelling, formatting and classification variations and errors than other published texts which have been edited in a more traditional and centralized manner. For applications such as this one which make use of Wikipedia as source data for statistical models, these types of errors are less significant, provided they are of relatively low frequency.

E. Creating a plain text corpus from Wikipedia

Users interact with Wikipedia as a set of article web pages. Each page contains global navigation links, links to article metadata such as the history and discussion pages, links to other articles within the article body text, and reference information such as footnotes.

To create a plain text corpus, only the actual body text is of interest. For continuous speech recognition language modelling purposes where the model should be trained on sentences approximating how people speak, punctuation and references are unwanted, and so further text conditioning (the process of converting text to a consistent, canonical form) is applied to transform wiki mark-up into a list of unpunctuated, upper-case sentences. Once a corpus has been created, it can be used to create a dictionary and a language model.

III. Goals for the adapted language model

When creating a custom language model adapted to a specific topic, the goal is not necessarily to create a larger model, but to create a well-adapted model, that is a model which is closely aligned to the recognition target text in genre, vocabulary, linguistic style and other dimensions. The size of an n-gram language model is initially determined by the number of different n-grams (combinations of n words) encountered in the training text. Models may be limited in size by:

- constraining the vocabulary, in which case words in the training text which are not in the given dictionary will be modelled as "unknown", and
- applying a frequency cut-off to the n-grams, in which case n-grams which occur fewer than a certain number of times in the training text will not be included in the model.

In general, a larger language model increases the search space for the recognizer, and for the Sphinx4 recognition engine, larger models lead to an increase in both runtime (a consequence of the larger search space) and memory requirements (a consequence of needing to load the entire model into memory).

A further consequence of increasing the search space with a larger model is that accuracy can be reduced as the model leads to the recognizer introducing extraneous words and phrases.

To enable the most accurate comparison between recognition performance with the adapted and reference language models, the adapted models were created with the same vocabulary size as the HUB4 reference model, 64000 words. However, owing to limitations in the language modelling toolkit used, a frequency cut-off was not applied to the adapted language model. This led to the adapted model having a larger number of bi-grams and tri-grams than the reference model, and overall being about twice the size. This may have contributed to reduced accuracy for our language models.

IV. Constructing a topic-adapted language model

Two types of topic-adapted language models were built. The difference between them is in the web crawler used to harvest Wikipedia articles. Two distinct corpora arise from these two sets of articles, and each corpus in turn produces a different language model. We refer to the two web crawlers as the Naïve Crawler and the Similarity Crawler.

A. Naïve Crawler

The operation of the naïve crawler can be broken down into 6 steps.

In Step 1, a Wikipedia search is executed using the Wikipedia Search API, and the first five articles meeting the

minimum seed article word count are added to the search queue (Step 2). The crawler then processes the article at the top of the search queue (Step 3). In Step 4, the markup text is conditioned to produce a list of plain text sentences, which is appended to the sentence output file. In Step 5, the markup text obtained in Step 3 is parsed to extract the set of links to other articles not already visited, and in Step 6, the titles of articles not already visited are added to the search queue. Steps 3 to 6 repeat until an exit condition is met. Exit conditions are: the maximum search depth has been reached (at most 5 links from the seed article to the indexed article), the maximum number of articles to index has been reached (2,500 articles), or the maximum number of output sentences has been reached (200,000 sentences). The full text of the article is retrieved in wiki markup format.

B. Similarity Crawler

When adapting a language model to a given topic, two goals are: (1) to improve the vocabulary coverage of the model for the given topic, i.e. to include as many words as possible which are likely to be used in the context of the topic, and (2) to model the style of language and typical word combinations used in the context of the topic. It is therefore advantageous to collect as much text as possible from contexts (here, Wikipedia articles) which are related to the target topic. And as this process should be unsupervised, it must be possible to establish "relatedness" in an automated way without subjective human judgement or interpretation.

This section describes an article similarity metric which gives the degree of similarity (measured from 0 to 1) between two Wikipedia articles. This metric is then used to improve the discrimination of a Wikipedia article crawler, such that only similar articles are included in the links which are followed. This approach, described further below, aims to gather a large set of articles using search seeding and transitive similarity.

A search is seeded using keywords, with subsequent articles being included in the search net through similarity to the parent article: The text from all such articles is then used to train a language model for the target topic.

Latent semantic indexing (LSI) was used to derive an article similarity metric. LSI, also known as Latent Semantic Analysis (LSA), is a technique widely used in information retrieval applications to identify related documents in large corpora [14], [15]. LSI uses singular value decomposition to train a model from the corpus which relates individual words to a set of topics. The set of topics is of a fixed-size and arbitrary (in that the topics are mathematical abstractions which emerge from latent semantic clustering in the data). Each topic is defined through a set of words and their respective contribution weights to the topic.

Using the model, a document may then be expressed as a set of topic values (representing the relative strength of each topic in the document), or equivalently as a set of n values representing a position in n-dimensional space (where n is the number of topics in the model). The similarity between two articles is then understood to be the distance between the two article vectors in an n-dimensional space.

To apply LSI to Wikipedia and generate article similarity scores, the open source gensim vector space modelling toolkit was used [16]. The gensim toolkit is designed to handle large corpora such as Wikipedia which exceed available memory, and in addition is well-documented and actively maintained.

The initial process to train the LSI model from Wikipedia is a modified version of the recipe described in the gensim documentation [17]. This requires three passes through an offline dump of all Wikipedia articles (3,345,476 articles in total from the Wikipedia snapshot used). This is a time-consuming process, but only needs to be executed at the start (and possibly at intervals thereafter to take account of gradual evolution of the corpus).

In Pass 1, a vocabulary is created of the most frequent 100,000 words. Only these words will be regarded as significant for LSI modelling, with any remaining words or tokens being ignored. Outputs from this pass are the list of words (each with a numeric identifier), and list of article titles (also each assigned a numeric identifier).

In Pass 2, a bag-of-words representation of each article is generated. This represents the article as the set of distinct words from the chosen vocabulary which occur in it. The output of this pass is a sparse matrix of words by articles.

In Pass 3, the sparse matrix is used to create the LSI model for 400 topics.

The model parameters of 100,000 terms and 400 topics followed the gensim defaults, informed by empirical results on dimensionality for semantic indexing applications suggesting an optimal range of 300 to 500 for topic size [18].

While the initial creation of the LSI model from Wikipedia was time-consuming (upwards of 24 hours), calculating similarity between a document and the set of documents to which it links was relatively efficient, at approximately 72 ms per comparison. This makes it a computationally tractable approach for generating custom language models on demand, requiring neither a significant memory footprint nor long runtime. By comparison, pre-computing pair-wise article similarity for the approximately 3.3 million articles in the Wikipedia snapshot would require a set of 5.6×10^{12} tuples, which would take just under 13,000 processor-years to calculate.

The Similarity Crawler differs from the Naïve Crawler in the addition of the Similarity Scorer in Step 6. Here the crawler passes a list of articles to the scorer, which calculates and returns a set of similarity scores for the target articles, ranging from 0 (least similar) to 1 (most similar). The crawler then discards articles which are insufficiently similar to the parent article.

The similarity threshold applied is $0.7 + 0.025 \times articledepth$. Thus articles linked from depth 0 articles (those returned by the keyword search) need to have a similarity of at least 0.7 to be included in the index queue, whereas for links from depth 1 articles, a threshold of 0.725 is applied, and so on. This is intended to counteract topic divergence as distance from the seed articles increases.

C. Building and using the language models

The following are the steps taken to construct a topic-adapted language model from Wikipedia.

1) A crawler harvests a set of articles (as described above) from Wikipedia which relate to the topic keywords.
2) The output text from each of the articles is conditioned into plain text sentences.
3) A target vocabulary is created for the adapted language model. This is done by merging two frequency-ranked vocabularies: one, the more specialized, derived from the output text from the selected set of Wikipedia articles, and the other, more general, derived from a plain text corpus of all Wikipedia articles. The merged list starts with all words which occur 5 or more times in the specialized word list, and is supplemented with words from the general list in descending order of frequency until the list reaches the target size of 64,000.
4) A phonetic dictionary is created for the target vocabulary, described further below.
5) The adapted language model is then created using the mitlm language modelling toolkit. As the amount of training text available from the set of topic-related Wikipedia articles is relatively small, a more general language model is first created from a larger Wikipedia corpus, restricted to the target vocabulary. The input corpus used for this model is 5% of all Wikipedia text, selected using 1 from every 20 sentences, yielding a total of around 75 million words. A topic-specific language model is then created from the conditioned text output from the topic-related Wikipedia articles, again restricted to the target vocabulary. The two language models are then merged using linear interpolation to create the third, adapted language model.

V. Pronunciation of unusual words

The base phonetic dictionary used is the CMU Pronouncing Dictionary (CMUDict) 0.7a, which contains phonetic representations for slightly over 123,000 words. However, it is to be expected that new words will be encountered which are not in the CMU Dictionary, for example because of their relative scarcity, or because they are neologisms or variants of known words such as new hyphenations.

As less common words are expected to be significant to the topic, it is important to recognize them where possible, and thus a method is required to generate pronunciations for unknown words. For this project, the phonetisaurus grapheme-to-phoneme (g2p) converter is used. Phonetisaurus uses a weighted finite state transducer (WFST) approach to generate pronunciation hypotheses for a word, a technique claimed to produce results comparable in accuracy to other state-of-the-art systems [19]. The model used by phonetisaurus for this application is trained from CMUDict 0.7a and thus phonetisaurus is in effect extrapolating from the implicit pronunciation rules represented in CMUDict. Only the best hypothesis generated by phonetisaurus is used. Further details on the use of phonetisaurus can be found in [20].

VI. Experiments and results

In order to evaluate the language models produced by the Naïve Crawler and the Similarity Crawler (we will refer to these as the Naïve LM and the Similarity LM), speech recognition was performed on the 13 lectures using the three types of language model and the accuracy of the transcripts were compared using various metrics discussed below. Fig. 1 shows the recognition process using the HUB4 language model, used as a reference language model in order to establish baseline performance. Fig. 2 similarly shows the recognition process using the custom language models.

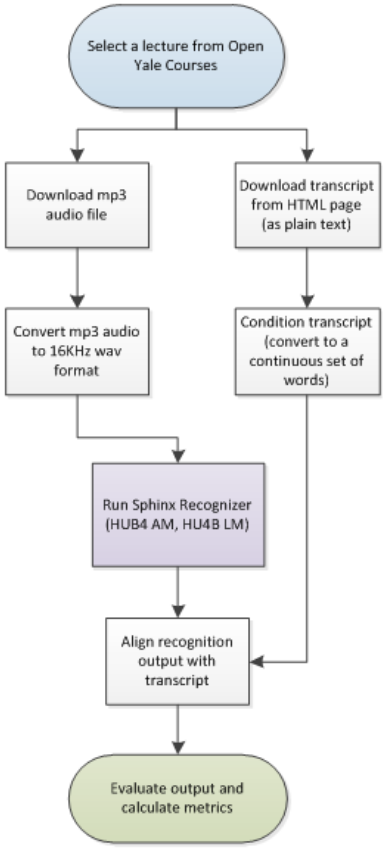

Fig. 1. Recognition with reference model

The method of language model adaptation here is unsupervised adaptation based only on minimal information about the lecture, in the form of up to 5 keywords derived from the lecture topic. It is assumed that a suitable set of keywords could always be selected, possibly in an automated way, from the subject area of the lecture (for example from the name of the department and title of the course) and the title of the lecture.

A. Metrics

The following metrics were used:

OOV Words is the number of out of vacabulary words, that is, words in the generated transcript which are not in the dictionary. Lower is better.

Perplexity measures a language model's alignment to the target text. Lower is better.

Word Error Rate (WER) is the number of insertions, deletions and substitutions required for the hypothesis to match the reference transcript, as a proportion of the

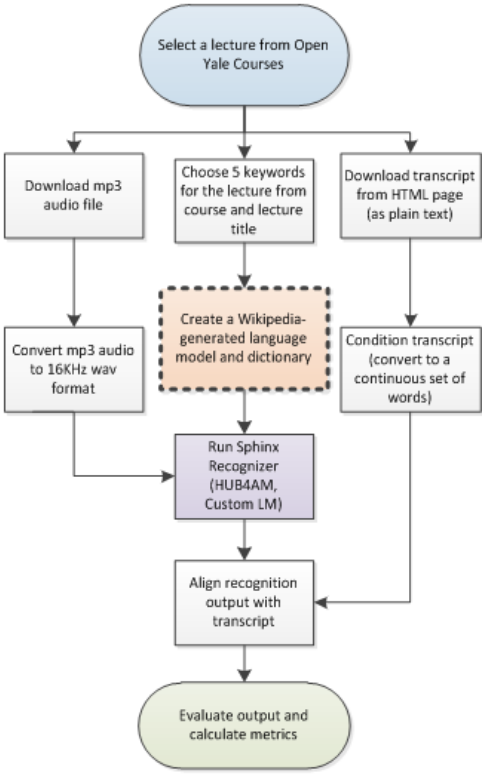

Fig. 2. Recognition with custom language model

TABLE I. LANGUAGE MODEL PERFORMANCE COMPARISON

LM	OOV	Perp	WER	WCR
Naïve	791	297	41.9%	68.0%
HUB4	1007	324	40.8%	71.7%
Sim	735	248	41.6%	68.4%

number of words in the reference transcript. Lower is better.

Word Correct Rate (WCR) is the number of words correctly recognized as a proportion of total word count in the reference. WCR ignores insertions. Higher is better.

B. Baseline Performance with HUB4

The HUB4 language model is used as the reference model (along with the HUB4 acoustic model). The HUB4 language model contains 64,000 unigrams. CMUdict 0.7a is used as the baseline pronunciation dictionary. Pronunciations have been estimated for 177 words which are contained in the HUB4 language model but are not in CMUdict.

Both measures indicate relatively wide variation in accuracy, with WER ranging from 31.8% to 61.1% and WCR ranging from 58.4% to 81.4% (Table I only shows averages due to space limitations). Audio factors which could account for this variation include the degree of background noise and reverberation in the recording (determined by room acoustics and microphone position), and the extent of alignment between the acoustic model and the speaker's accent.

A difficulty in examining the impact of changes in the recognition process is in understanding the extent to which acoustic or language factors dominate recognition accuracy. Nevertheless, the average WER here of around 40% is consistent with reported results from other projects and recognizers ([21], [22]).

C. Performance of Naïve and Similarity LMs

To investigate whether the Similarity Crawler produced better language models than the Naïve Crawler, the recognition performance of the language models derived from the respective Wikipedia crawlers was compared across the four metrics: OOV words, perplexity, WER and WCR. Table I shows that the Similarity LMs outperform the Naïve LMs across all metrics, although in some cases by a relatively small amount. Vocabulary coverage improved, perplexity was lower, WER was better though by only 0.3%, and WCR was better by 0.4%.

D. Performance of HUB4 and Similarity LMs

Having seen that the Simlarity LMs were better than the Naïve LMs, it remained to investigate whether the Similarity LMs were also better than the baseline LM. Table I shows that on average the Similarity language models outperformed HUB4 in the two language-related metrics (OOV and perplexity), but recognition performance for the Similarity LMs reflected in WER and WCR was actually worse, with an increase in WER of 0.8% and a decrease in WCR of 3.3%.

A possible contributing factor to the degraded recognition performance is estimated pronunciation. As an aim of the Wikipedia crawler is to introduce specialist vocabulary into the topic-adapted language models, it is likely that a number of such words will not occur in the relatively small CMU pronouncing dictionary (around 123,000 words). Pronunciations for such words are therefore estimated, in this project through the phonetisaurus grapheme-to-phoneme tool using a model trained from CMUdict [19].

As these estimations are extrapolations of implicit rules in CMUdict, they may be inaccurate for unusual or foreign vocabulary and thus produce poor recognition results. For example, the word "Lycidas" from the lecture on Milton was examined. The estimated pronunciation "L AY S AH D AH Z" resulted in zero recognition rate whereas with the manual pronunciation "L IH S IY D AH S", 7 out of 47 instances were correctly recognized. Futher investigation confirmed this difficulty. For CMUdict pronunciations, 81% of all words were recognized at least once, whereas only 46% of words with estimated pronunciations were ever recognized correctly. Estimating pronunciation was therefore only partially effective.

E. Searchability and Ranked Word Correct Rate (RWCR)

The results seen earlier show that the topic-adapted language models have a net negative effect on WER and WCR. The resulting transcripts are therefore likely to be less readable. However, in relation to the goal of improving searchability, not all words are created equal: users are more likely to use less common words as search terms. Do the topic-adapted language models therefore lead to more searchable transcripts, or, defined in information retrival terms, provide better recall?

To answer this question, word recognition performance was examined across four word rank frequency groupings, using a

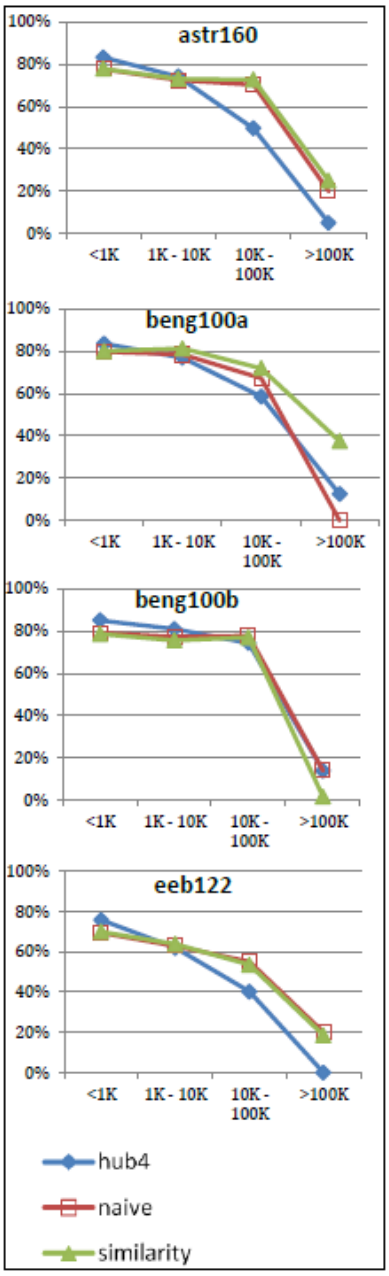

Fig. 3. Partial Word Correct Rate by word frequency rank groups

1.5 million word-frequency dictionary derived from English Wikipedia (words of 3 or more characters ordered from most to least frequent).

Fig. 3 presents the Word Correct Rate for words in each frequency rank group for 4 lectures by language model. For example, the WCR for 1K–10K is given by calculating the word recognition rate for all words ranked from 1,000 to 10,000 in the English Wikipedia frequency-ranked dictionary. The HUB4 language model (shown in blue) outperforms the Naïve (red) and Similarity (green) LMs in all cases for words under rank 1000; from 1K to 10K results are similar, whereas for 10K–100K and above, the Naïve and Similarity models outperform HUB4 in 3 out of the 4 cases shown here.

Examining recognition accuracy by word frequency rank therefore seems helpful in providing a better characterisation of the performance of topic-adapted language models in recognizing less common words. To simplify such analysis, a single metric is proposed: Ranked Word Correct Rate (RWCR-n), defined as the Word Correct Rate for all the words in the document which are not found in the first n words in a given general English word dictionary with words ranked from most to least frequent. At n = 0, RWCR is identical to WCR and may diverge as n increases.

TABLE II. LANGUAGE MODEL PERFORMANCE COMPARISON WITH RWCR-10K

LM	OOV	Perp	WER	WCR	RWCR-10k
HUB4	1007	324	40.8%	71.7%	46.0%
Sim	735	248	41.6%	68.4%	54.9%
Diff	272	76	0.8%	3.3%	9.0%

Table II shows the performance of the HUB4 and Similarity LMs across all lectures for the four metrics seen previously and for RWCR-10K.

RWCR-10K improves by 9% from the HUB4 to Similarity LM, even though WER worsens on average by 0.8% and overall WCR worsens by 3.3%.

Using the RWCR-10K metric, it appears therefore the topic-adapted language models are successful in improving recognition of less common words, although they do so at the expense of recognition of more common words and thus overall accuracy.

VII. CONCLUSION AND FUTURE WORK

We have described a project which explored the usefulness of topic-adapted language models in increasing the searchability of automatically generated transcripts of academic lectures. The adaptation was performed using Wikipedia as a source corpus. Transcripts produced using the topic adapted language models were compared to those produced using the HUB4 language model, used here as a reference model. While the standard metrics of Word Error Rate and Word Correct Rate showed that the transcripts produced by our models were less accurate (hence less readable), our RWCR-10K metric, which is better aligned with the goal of searchability, showed significant improvement.

It is, however, important to produce transcripts that are both searchable and readable. Future work should look into how both these goals can be achieved. One problem we noticed in our study was that many of the unusual words we focused on did not have pronunciation encoded in CMUDict, the pronunciation model used by the recognition engine. Instead we estimated their pronunciations in an automated but rather crude fashion. There was evidence that this had a detrimental effect on recognition accuracy and in future should be dealt with by providing proper pronunciation for unusual words.

Another issue of concern is the use of Wikipedia, whose generality can be both helpful (because any subject can be expected to be found there) and harmful (because it makes it difficult to find what one needs and exclude what one does not need). This can be addressed by using scholarly repositories, such as journal archives, as a corpus source.

REFERENCES

[1] J. Copley, "Audio and video podcasts of lectures for campus-based students: production and evaluation of student use," *Innovations in Education and Teaching International*, vol. 44, no. 4, pp. 387–399, 2007.

[2] C. Munteanu, G. Penn, R. Baecker, E. Toms, and D. James, "Measuring the acceptable word error rate of machine-generated webcast transcripts," in *Ninth International Conference on Spoken Language Processing*, 2006.

[3] K. Kato, H. Nanjo, and T. Kawahara, "Automatic transcription of lecture speech using topic-independent language modeling," in *Sixth International Conference on Spoken Language Processing*, 2000.

[4] H. Yamazaki, K. Iwano, K. Shinoda, S. Furui, and H. Yokota, "Dynamic language model adaptation using presentation slides for lecture speech recognition." in *INTERSPEECH*, 2007, pp. 2349–2352.

[5] C. Munteanu, G. Penn, and R. Baecker, "Web-based language modelling for automatic lecture transcription," in *Eighth Annual Conference of the International Speech Communication Association*, 2007.

[6] T. Kawahara, Y. Nemoto, and Y. Akita, "Automatic lecture transcription by exploiting presentation slide information for language model adaptation," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, 2008, pp. 4929–4932.

[7] H. Bourlard, H. Hermansky, and N. Morgan, "Towards increasing speech recognition error rates," *Speech communication*, vol. 18, no. 3, pp. 205–231, 1996.

[8] I. McCowan, D. Moore, J. Dines, D. Gatica-Perez, M. Flynn, P. Wellner, and H. Bourlard, "On the use of information retrieval measures for speech recognition evaluation," *IDIAP (Institut Dalle Molle d'Intelligence Artificielle Perceptive), Martigny, Switzerland. IDIAP Research Report IDIAP-RR 04-73, March 2005*, 2005.

[9] A. Park, T. J. Hazen, and J. Glass, "Automatic processing of audio lectures for information retrieval: Vocabulary selection and language modeling," in *Proc. ICASSP*, vol. 1, 2005, pp. 497–500.

[10] Y.-Y. Wang, A. Acero, and C. Chelba, "Is word error rate a good indicator for spoken language understanding accuracy," in *Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on*. IEEE, 2003, pp. 577–582.

[11] R. Parker, D. Graff, J. Kong, K. Chen, and K. Maeda, "English gigaword fifth edition," *Linguistic Data Consortium, Philadelphia*.

[12] S. P. Ponzetto and M. Strube, "Knowledge derived from wikipedia for computing semantic relatedness." *J. Artif. Intell. Res.(JAIR)*, vol. 30, pp. 181–212, 2007.

[13] W. contributors. (2012, Jul.) Wikipedia:copyrights. [Online]. Available: http://en.wikipedia.org/wiki/Wikipedia:Copyrights

[14] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse processes*, vol. 25, no. 2-3, pp. 259–284, 1998.

[15] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *JASIS*, vol. 41, no. 6, pp. 391–407, 1990.

[16] R. Řehůřek and P. Sojka, "Software framework for topic modelling with large corpora," in *Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks*, 2010, pp. 46–50.

[17] R. Řehůřek. (2011, Dec.) Experiments on the english wikipedia. [Online]. Available: http://radimrehurek.com/gensim/wiki.html

[18] R. B. Bradford, "An empirical study of required dimensionality for large-scale latent semantic indexing applications," in *Proceedings of the 17th ACM conference on Information and knowledge management*. ACM, 2008, pp. 153–162.

[19] J. Novak, D. Yang, N. Minematsu, and K. Hirose, "Initial and evaluations of an open source wfst-based phoneticizer," *The University of Tokyo, Tokyo Institute of Technology*.

[20] L. Galescu and J. F. Allen, "Bi-directional conversion between graphemes and phonemes using a joint n-gram model," in *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*, 2001.

[21] J. R. Glass, T. J. Hazen, D. S. Cyphers, K. Schutte, and A. Park, "The mit spoken lecture processing project," in *Proceedings of HLT/EMNLP on Interactive Demonstrations*. Association for Computational Linguistics, 2005, pp. 28–29.

[22] C. Munteanu, R. Baecker, G. Penn, E. Toms, and D. James, "The effect of speech recognition accuracy rates on the usefulness and usability of webcast archives," in *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, 2006, pp. 493–502.