Natural Language Processing for Information Retrieval: the time is ripe (again)

Matthew Lease
Brown Laboratory for Linguistic Information Processing (BLLIP)
Brown University
Providence, RI USA
mlease@cs.brown.edu

ABSTRACT

Paraphrasing van Rijsbergen [37], the time is ripe for another attempt at using natural language processing (NLP) for information retrieval (IR). This paper introduces my dissertation study, which will explore methods for integrating modern NLP with state-of-the-art IR techniques. In addition to text, I will also apply retrieval to conversational speech data, which poses a unique set of considerations in comparison to text. Greater use of NLP has potential to improve both text and speech retrieval.

Categories and Subject Descriptors

H.3.3 Information Search and Retrieval [**Information Search and Retrieval**]: Retrieval models

General Terms

Experimentation

1. INTRODUCTION

The broad goal of information retrieval (IR) is to develop systems which can automatically provide relevant information corresponding to an expressed information need. If this information need and the relevant information sought are both expressed in human language, then success on this task ultimately depends on how well a system can model and understand language. In practice, deep understanding has remained elusive while shallow, bag-of-words style methods continue to dominate IR.

Roughly a decade ago, Karen Sparck Jones and several others independently wrote a collection of papers reflecting on the observed and potential contribution of natural language processing (NLP) to IR [15, 21, 33, 35]. Though work in this vein goes back nearly as far as IR itself, their bleak observation might be best summed up in Smeaton's remark that *the impact of NLP on information retrieval tasks has largely been one of promise rather than substance* [33]. Perhaps unsurprisingly, the years since their writing have seen a dearth of work in applying NLP to retrieval [3]. Nevertheless, further precision improvements within the existing paradigm have become increasingly marginal and difficult to eke out, making it increasingly clear that we must start thinking beyond the existing framework if we wish to achieve further substantial gains (and with precision typically below 50%, there is much room left for improvement).

To paraphrase van Rijsbergen [37], the time appears ripe for another attempt at using natural language processing for retrieval, to consider anew this line of research and see if any light may have been shed on it in intervening years. With regard to NLP, the statistical revolution has continued to expand the field's horizons; the field today is thoroughly statistical with robust methodology for estimation, inference, and evaluation. As such, one may well ask if there are new advancements that suggest re-exploring prior directions in applying NLP to IR? Are there promising avenues still unexplored? Is the utility of NLP for IR primarily restricted by the accuracy of recognizing its formalisms, or are the formalisms themselves insufficient? Questions such as these represent the core of the inquiry to be undertaken.

Many interesting questions also remain regarding speech retrieval. Around the same time Sparck Jones and others were commenting on the general failure of NLP to benefit IR, TREC's Spoken Document Retrieval track declared searching speech to be a "solved problem" [12]. It seemed that solving text retrieval would largely solve speech retrieval as well because retrieval methods were found to be remarkably robust to word recognition errors. Despite the track's impressive results, hindsight has shown that many interesting problems remain [1, 23]. For example, broadcast news (BN) data used for the track is quite different from conversational speech (CS) such as found in interviews, debates, meetings, classroom discussion, talk shows, telephone conversations, online chat, etc.. Word error rate is higher, topic segmentation is more problematic (potentially involving speaker identification and conversation untangling [5]), and indexing and use of retrieved content is complicated by back-channels, disfluency (filled pauses, explicit editing terms, self interruptions and corrections, etc.), and dramatically different sentential structure as speakers trail off, interrupt one another, and compose their utterances on-the-fly. These phenomena pose challenges to current speech retrieval techniques in addition to deeper NLP analysis.

These two distinct yet complementary lines of research comprise the thesis work: NLP-informed retrieval of both textual and conversational speech data. Methods developed will be compared to existing state-of-the-art practices and

PIKM'07, November 9, 2007, Lisboa, Portugal.
Copyright 2007 ACM 978-1-59593-832-9/07/0011 ...$5.00.

build on existing foundations in syntactic analysis of text and language modeling [6, 25], rich transcription and syntactic analysis of spontaneous speech [14, 19], and some preliminary work in retrieving spontaneous speech under the language modeling paradigm [17]. At this stage of formulating the dissertation, additional input will be particularly valuable in shaping its course with regard to important previous work to re-examine and alternative directions to explore.

One important issue to address from the outset is that of efficiency: performing any sort of analysis more sophisticated than collecting word counts will clearly involve more processing time than would a system which must only compute simple statistics. If one's goal is to quickly produce a competitive web scale search engine with instant response time (ala Google), there isn't much wiggle room for research to introduce any additional computation. As such, a very different efficiency goal is envisioned for this work than what is commonly seen in the IR literature: I intend to address efficiency only to the extent necessary to evaluate proposed methods on a reasonable-sized collection of documents, such as those used in TREC evaluations. Even if the techniques developed here could not be scaled to the Web in the near future, I would consider the research successful should it demonstrate that deeper understanding of documents and queries could provide significant new traction on the IR task. Such a result would be consistent with my overarching, long-term research interest in developing systems capable of more deeply understanding human language, and it's difficult to imagine that deeper analysis is not the direction we all ultimately want to move toward in the long run.

The task of text retrieval is introduced in §2 along with a brief survey of modern practice. In §3, I consider why past use of NLP has not been more successful and propose a few places previous work might be revisited and new avenues explored. Conversational speech data is described in §4 and introduces prominent phenomena differentiating conversational speech from text as well as their effect on retrieval methods. Discussion of a few evaluation conditions and their potential impact on NLP-based retrieval is found in §5, and §6 provides closing remarks.

2. TEXT RETRIEVAL MODELS

This section introduces the task of *ad hoc* text retrieval (TR) and briefly reviews current practices [24, 32, 38], giving particular attention to the feature space. In distinguishing TR from IR, the primary intent is simply to indicate the information being retrieved is human language rather than, say, video. However, an additional distinction is also being made here between textual and speech data, the latter of which is the subject of §4.

In ad hoc TR, the system is given a user *query* expressing an *information need* and a *collection* of documents in which to search for that information. The goal is to return a list of documents, ranked in order of (estimated) decreasing relevance, which the user may then peruse, use to refine his search, etc.. Assuming the availability of a set of "canned" queries and corresponding human relevance assessments over the collection, the accuracy of a given system can be empirically evaluated and its strategies refined. Though I focus attention on the ad hoc scenario, reflecting my interest and work to date, two other well-known task variants should be mentioned. In *routing*, the system is provided examples of relevant documents for a given query and must rank relevance of additional documents *on the same query*; for example, a user may indicate several documents relevant to his query in hopes of improving system accuracy in retrieving further documents. Another task variant known as *filtering* requires systems to make an independent relevance decision for each document and penalizes wrong decisions; for example, a user subscribed to a news feed may wish to automatically delete on arrival any story outside his area of interest. Historically, filtering has often assumed the availability of example relevant documents as with routing, but my use of the term excludes this assumption. Unless otherwise mentioned, I will use TR to refer to the ad hoc scenario.

The classic and still competitive approach to TR is to represent the query and documents as vectors over the collection vocabulary and rank documents on the basis of vector similarity [32]. This approach begins by assuming words to be independent of one another: no attempt is made to model inter-word relationships in either the query or the documents. While no one would argue this assumption is actually valid in human language, it simplifies modeling and has performed remarkably well in practice despite its naivety. Given this assumption, statistics are collected for individual terms in the query and documents, and document relevance is computed by summing a term-relevance function calculated for each query term. Several key statistics are involved in computing this function: term frequency (TF), inverse document frequency (IDF), and length normalization. TF is a measure of term salience: the more often a query term occurs in a document, the more information the document is assumed to contain relevant to that term. Since terms will tend to occur more frequently in longer documents regardless of topic, document length normalization is usually applied to remove this bias: relative term frequencies are used in place of absolute counts. IDF measures term importance: a query term occurring rarely in the collection is assumed to be more important in discriminating between documents than a determiner like "the", which likely occurs in every document. A major and widely-adopted advancement over basic TF-IDF is pivoted document length normalization, which applies variable (non-Euclidean) normalization to correct for observed error between estimated relevance under standard normalization and relevance values observed on development data [32, 9].

Like the vector-similarity approach, the *probabilistic* approach to TR also has a long history and performs competitively. This approach is based on Robertson's famous probability ranking principle (PRP), which showed that optimal system behavior could be achieved by ranking documents according to the probability of their belonging to the relevant class (i.e. assuming binary relevance/non-relevance, documents can be classified as belonging to either one set or the other) [30]. Rather than compute vector similarity, probabilistic systems directly estimate the probability of relevance for each document. In practice, successful systems within this paradigm have made use of roughly the same TF-IDF statistics used with vector similarity. The most famous probabilistic TR system is Okapi BM25, which has been refined over many years of participation in TREC evaluations [36, 9]. In addition to the basic TF-IDF statistics, BM25's probability model also incorporates average document length, provides several free parameters for tuning on development data, and facilitates query term weighting, which has been shown to be useful with longer queries. In a routing scenario, an expansion of the formula allows examples of relevance to be exploited, achieving a tighter connection to the PRP.

A more recent approach to TR has been developed based on language modeling [29]. In this paradigm, one assumes a unique language model (LM) underlies each observed document and estimates document relevance by the probability of observing the query as a random sample generated by the document's underlying LM. Usually one assumes bag-of-words independence similar to that employed with the probabilistic and TF-IDF models: the probability of a string of words is computed as the product of the individual word probabilities (i.e. a unigram model). One challenge of the LM paradigm is estimating the parameters of the underlying LMs given the brevity of the observed documents; if one simply takes the maximum likelihood estimate (MLE), a single query term unobserved in the document would zero-out the entire probability of observing the query given the document, making the entire framework exceedingly fragile. Instead one commonly employs smoothing to discount the probability mass assigned to observed terms and reserve some probability mass for all unseen terms. Assuming such smoothing is employed and one adopts a uniform prior over documents (a standard assumption that all documents are equally likely to be relevant before seeing the query), the LM approach has been shown to have a strong theoretical connection to TF-IDF [39] and perform comparably to the other two approaches in practice [9]. A potential advantage of the LM approach lies in the pre-existing theoretical foundation and set of proven estimation techniques developed by earlier work in speech recognition.

Input queries are often non-optimal; information is often lost in translating the user's information need into a system query, and there is usually a significant paraphrase mismatch in how the query and relevant documents refer to the same information. Consequently, a simple strategy for improving all of the word-based approaches discussed above is to augment the original query with additional terms harvested from known relevant documents, as are available in the routing task. Of course, were such known relevant documents available, one might also adopt a very different approach to retrieval: the PRP may be more directly applied in the probabilistic approach, and in the LM paradigm, one could assume an LM underlies the relevant class and use the relevant documents to directly estimate its parameters [16][1]. A challenge of this strategy, however, is convincing a user to invest the additional time required to identify a set of such documents relevant to his query. Fortunately, a bootstrapping/semi-supervised strategy has been shown to perform quite well in practice: one runs any text retrieval system to get a preliminary ranking of documents, assumes some number of the top-ranked documents are indeed relevant, and expands the original query using their terms [36]. While the performance gain achieved by pseudo-relevance feedback has been shown to vary greatly with parameters employed (how many documents to assume are relevant, which terms to harvest, whether to iterate feedback, etc.), the improvement is usually quite substantial.

By machine learning standards, all of the approaches discussed thus far are rather spartan with regard to the feature set employed. While introducing additional model parameters should not be a goal in and of itself (requiring greater manual tuning or training data for automatic estimation), adding parameters can be useful to the extent it allows one to model and leverage additional correlations and structure in the data. With the predominantly term-specific features used today, it is difficult to learn effectively from past experience: knowing a given document is relevant to a given query, how do we generalize this knowledge to improve accuracy on future queries? The routing task only considers applying such knowledge to the *same* query, and so sidesteps this question. While the approaches discussed earlier do tune a few parameters on development data, by and large they make little attempt to model query-independent correlation between queries and relevant documents. Prior work applying machine learning within this limited feature space has had somewhat limited success [24, 38]. To improve upon today's state-of-the-art, we should look toward incorporating a broader range of features under an effective learning framework. A generative approach would maximize the joint likelihood of queries and their relevant documents, whereas a discriminative approach would maximize precision in partitioning the collection for relevance. Discriminative models are potentially advantageous in both avoiding metric divergence and being able to exploit negative examples as well as positive ones [26], while (Bayesian) generative models allow evidence to be easily incorporated and avoid slow partition function computation. There is a history in using both types of models for retrieval [24, 26, 38], and both have been used successfully for a variety of tasks. Though recent work has studied learning frameworks in detail [26], the potential feature space for TR remains largely unexplored.

3. TEXT RETRIEVAL AND NLP

Despite the fact that work applying NLP to text retrieval (TR) goes back nearly as far as TR itself, NLP has not had a significant impact on TR to date [3, 35]. As such, prior to embarking upon another such attempt it is worthwhile considering why previous work was not more successful and what makes the present outlook more promising? This section presents a preliminary and cursory look at these questions. Initial discussion is framed in terms of a recently proposed conceptual model for incorporating NLP into TR [40]. This is followed by a more concrete discussion of a few avenues by which use of NLP might yield tangible benefit. In comparison to previous work, the intent here is to explore novel directions with regard to the feature set used, techniques for more effectively extracting those features, and strategies for exploiting them in TR.

The conceptual model proposed [40] has a widely-known analog in machine translation (MT) by which I will introduce it. In the MT model, three corners of a pyramid represent an information item in three alternate forms: at each corner of the pyramid's base the information is represented in a human language, and at the apex, an interlingua representation (i.e., deep meaning). The task of MT is then interpreted as trying to get from one corner of the base to the other (i.e. to transform one language's string of words into the other's), and the question is which path to take to best realize this transformation? The most shallow approach tries to directly map between each language's surface words, whereas the deepest approach attempts to connect

[1] Given the estimated class model, inference in the LM paradigm is then performed to determine the likelihood of the class having generating a given document; this is in contrast to the original inference problem of determining the likelihood of the query string given the document's LM.

the languages via interlingua. Between these two extremes lies an infinite range of design choices: the source and target languages can be analyzed to arbitrary depth (with different choices possible for each) before attempting to map between them. Returning to TR, the conceptual model proposed is a similar pyramid: one can choose independently how deeply to analyze the query and documents before seeking correspondence between their representations. Whereas state-of-the-art methods today operate primarily at the level of surface-level words, one would hope that that deeper analysis of some form could help address some of the paraphrase challenges from trying to directly match surface strings [21]. In such a framework, the lines between information extraction (IE) and IR would begin to blur as IE were increasingly employed to extract deeper representations for matching [40]. While work in statistical MT began at the pyramid's base, recent years have moved toward deeper representations, and one might hope to see a similar trend in TR.

3.1 Syntax

Given the bag-of-words independence assumption underlying the TR approaches (§2), modeling some notion of how words actually relate to one another seems like an obvious first step toward developing a richer representation of queries and documents. Previous work in this vein has focused primarily on identifying informative word pairs or longer phrases as larger units for matching. A natural source of such candidate phrases arises from linguistic theories of syntax, and use of syntax for TR goes back to the 60s [35]. While use of non-linguistic phrases has been shown to yield around 10% relative improvement [3], linguistic phrases have provided limited additional benefit [3, 21, 32, 33, 35].

So why is there still reason to believe that modeling syntax could be useful? Syntax informs us about the compositionality of human language: the infinite ways words combine to form phrases, and how those phrases in turn combine to eventually form sentences. Most syntactic theories define a systematic relationship between syntactic structure and meaning, and at minimum it seems we must know which words modify which other words to get the correct interpretation. Consequently, accurate recovery of syntax (i.e. parsing) is widely viewed in NLP as a necessary precursor to building systems capable of understanding natural language. While the holy grail of fully understanding language remains largely elusive, parsing has in the meantime been usefully applied to a variety of practical tasks [18].

Previous work in applying syntax to TR can be usefully characterized in terms of the following: syntactic theory adopted, accuracy in automatically recovering its representations, and how those representations were employed. Of the three, I think the choice of theory is probably the least critical so long as its syntax can be automatically recovered with reasonable precision. Subtle differences between competing theories are likely beyond what we can effectively detect and exploit with today's models. Most work in statistical parsing today has adopted the theory implicit in the Penn Treebank (PTB), about two million words from newspaper text and telephone conversations manually annotated for syntax [22]. While PTB's theory may not be perfect, the annotation standard is fairly rigorous and captures most syntactic phenomena of practical interest.

The second point mentioned above was parsing accuracy. Prior to the recent statistical revolution in NLP, parsing methods suffered from poor coverage and had difficulty resolving ambiguity. Today's parsers are far better on both counts: the state-of-the-art Charniak parser can correctly match 92% of manually-annotated syntactic constituency in PTB's newspaper text with full coverage [25]. And today's statistical models are remarkably robust across "non-standard" language usage like conversational speech (even in presence of word and sentence boundary recognition errors) [14] or sub-sentential phrases like scientific paper titles or TREC-style topic titles and descriptions [17]. Syntactic language models have also been shown to both outperform and operate synergistically with n-gram models and can be further improved with additional un-annotated data [6]. Of course, accuracy is a moot point to the extent we can upper-bound a given TR scheme using manually-annotated syntax – that is, to evaluate the utility of the formalism separate from our ability to automatically recognize it. While it would not be feasible to do this across an entire collection of documents, one could manually annotate syntax for some canned queries. I am not familiar with any previous work reporting use of manually annotated syntax.

The third and arguably most important point is the question of how a given syntactic representation might be best exploited in order to improve retrieval accuracy. While previous work has focused on simply identifying word pairs or phrases for matching, there is a much larger space of possibilities. One simple, untried idea would be to consider part-of-speech tags and syntactic heads in determining term weights [3]. Rather than just identifying related words, one could model attributes of the relationship: phrasal category, modifying direction, element types, etc. Going beyond syntactic neighbors to indirect relations like ancestry, further structural correlation may be sought. As the number of features grows, so too will the challenge in determining effective feature weights, particularly given the brevity of queries and documents. One strategy for addressing this may be learning feature weights in a discriminative framework, as has been effectively applied with *millions* of features to parse re-ranking [18]. Such a framework may also be useful in facilitating convenient exploration of the feature space [26, 38]. Greater exploration of the syntactic feature space may reveal that the value of syntax is most pronounced when used in combination with TF-IDF statistics and other semantic features as part of a joint framework (which might model statistically-induced phrases as well).

My initial work will adopt the LM paradigm for TR and compare the relative effectiveness of bigrams [34] vs. syntactic bi-lexical dependencies as recovered by Charniak's state-of-the-art parsing model [25]. The goal is to revisit previous work [10, 20, 27] with a more accurate parser. Although Gao et al. [10] hypothesize the strength of their system is non-syntactic, their cited examples indicate their retrieval accuracy was most improved by syntactic dependencies, suggesting more accurate parsing could help. I have also begun work in applying Charniak's syntactic LM [6] to retrieval.

3.2 Co-reference and named-entities

People often use different descriptors to refer to the same entity or idea. The simplest example of this can be seen with pronouns, in which he/she/it are used to refer back to entities introduced earlier in the discourse. Another example would be referring to the same person via different forms of their name, their job title, or other personal characteristics, etc.. Use of co-reference has clear implications for term matching strategies in IR, since multiple references to the same entity in a query or document would be mistakenly matched by current methods on the basis of the referring terms rather than referent entities. Whereas use of co-reference resolution techniques have been recently reported on tasks like question answering and summarization, modern techniques and/or manual annotations have seen little use in recent TR literature.

While co-reference resolution can inform TR by identifying which words refer to which entities, knowledge of named-entities (person, place, organization, etc.) may be further exploited to improve search. Greater weight could be assigned to entities than non-entities when matching, and one could further adjust weights between entity types, perhaps given evidence regarding the distribution of types in the query or using stronger evidence inferred regarding the type of information sought. Some limited use of named-entities has been recently reported [11], though their individual contribution versus that of other components is difficult to gauge without leave-one-out feature analysis.

3.3 Lexical similarity

As discussed in the context of pseudo-relevance feedback (§2), query expansion is a powerful method for addressing both (1) loss of information in the user's translation of their information need into their generated query and (2) paraphrase mismatch between the query and relevant documents' description of the same information. While pseudo-relevance feedback addresses these issues in terms of the collection and current query, lexical similarity schemes are often conceived of more generally. For example, thesauri are often manually constructed to describe restricted, well-defined notions of similarity such as synonymy. A looser notion of relatedness can be found in word-similarity methods, which are inherently statistical and effectively defining a mapping from every pair of words to a real-valued similarity score. The benefit of applying each approach to TR can be broadly distinguished as addressing synonymy (thesauri) versus expressing related concepts (word similarity), but there is a large gray area between these two extremes. In both cases, standard practice is to expand the query with additional, weighted terms and then to match surface forms between query and document as usual. Both approaches have been applied recently with some success [2, 4].

While manual thesauri present several challenges in terms of limited coverage and inferring probabilities governing the specified symbolic relations, some recent work has explored directions for acquiring such relations broadly and statistically from the web [8]. The idea is that strong probabilistic cues of well-defined relations like synonymy, hypernymy (kind-of), and meronymy (part-of) can be learned automatically by complementing shallow analysis with massive redundancy, and initial results look promising. A similar strategy can be adopted for acquiring word similarity information, and both approaches merit further investigation.

Word clusters provide another potentially useful resource for lexical similarity. Rather than define precise term relationships like thesauri or word-pair numerical similarity, they define a notion of concept groups that has been successfully exploited in previous language modeling work [13]. Such use of word clusters can be naturally studied within the LM paradigm for TR and will be part of my work.

3.4 Text normalization

Discussion of a deeper representation for text often centers on an incomplete notion of interlingua that could not be broadly realized or a placeholder for some similar future breakthrough. However, much simpler representations can be broadly realized today as a compromise between observed surface structure and hypothesized interlingua. To give a simple example from syntax, topicalization or fronting (i.e. *Yoda-speak*) may be easily undone to render the sentence in a normalized format for subsequent analysis. Another example is that speech data is often normalized to be more like text in order to leverage text resources or apply text-based analysis tools. The idea of normalization becomes potentially more interesting as we move from simple syntactic reordering toward a deeper representation like predicate-argument structure in which more details of surface form realization are stripped away. While such structure does not help with recognizing world-knowledge style paraphrase (e.g. the president of a company being its *head*), it may nonetheless help close the gap recognizing paraphrase in some alternative surface realizations of similar information. It seems a useful place to begin considering this for TR is to examine the question answering (QA) literature: recognizing sentential paraphrase is a more central task in this community in order to match questions and answers, QA training data provides a useful place to start comparing the utility of different normalization schemes, and there is some reason to believe success in recognizing sentential paraphrase should provide leverage in recognizing paraphrase between queries and documents for retrieval.

4. RETRIEVING SPONTANEOUS SPEECH

TREC's Spoken Document Retrieval (SDR) track is considered one of the great successes of the TREC program, having declared nearly a decade ago that searching speech was a "solved problem" [12]. The general finding was that while automatic speech recognition (ASR) was far from perfect, text retrieval methods were remarkably robust in the face of word recognition errors; assumably the words which were correctly recognized provided sufficient evidence to overcome the random noise introduced by the misrecognized words. However, there are a couple potential caveats to this finding worth considering. One is that broadcast news (BN) (used in the track) is fairly redundant: the same information is often repeated several times in different ways, so a retrieval system may have several opportunities to correctly recognize a term of interest. Use of longer queries has similarly compensated for recognition error by providing more terms for matching [23]. BN also presents information in an organized format which lends itself well to being chopped up into small, cohesive segments for independent retrieval (leaving the problem of automatic segmentation to be solved separately). This issue of segmentation has implications for robustness to word recognition errors: longer, cohesive segments provide more related words to compensate for misrecognized ones [1]. Overall, BN represents perhaps the closest approximation of text by speech except read documents.

Conversational speech (CS), as found in interviews, debates, meetings, classroom discussion, talk shows, and telephone conversations, is a surprisingly different form of language when studied closely. With regards to retrieval, the most noted difference in the literature is higher word error rate (WER): there is a broader class of speakers and accents, often poorer recording conditions, greater use of dialectal speech and code-switching (mixing of dialects and/or languages). Perhaps less obvious, differences in discourse structure, style, and syntax with comparison to text reduce the ability of text-trained language models (LMs) to accurately distinguish between equally plausible acoustic alternatives. While WER has certainly improved in recent years, the state-of-the-art for CS is still significantly worse than for BN [23]. Named-entities, often of interest for the purposes of retrieval, are more likely to be out-of-vocabulary since the general almanacs, name and company lists, etc., used in building recognition models are less likely to cover entities discussed in CS than in BN. Most work to date in SDR has assumed 1-best recognizer output, as this allows text-based retrieval systems to be easily ported to speech, but this assumption is limiting as WER grows. In contrast, strategies like phoneme-based retrieval or working off word lattices or n-best recognizer output allows greater flexibility in compensating for 1-best recognizer error [31].

Segmentation of CS into clearly demarcated topics, as done with BN, can also be challenging since conversational topics are often open ended and topical threads may freely wrap back and forth and intertwine one another. A cocktail party or online chat forum make coherent segmentation even more difficult since multiple conversations take place in parallel. Whether or not parallel discussion occurs, there is often a high degree of overlap between speakers actively engaged in a conversation such that literally transcribing the conversation recorded by a single microphone can produce something resembling word salad (the chat forum equivalent is sentence salad). Better recording conditions can certainly help alleviate the problem with speech, but a general retrieval system for CS may not be able to rely upon this. Thus in addition to topic detection, which as already noted is already more difficult for CS than BN, we have two additional challenges in segmentation: automatic speaker identification (if recording conditions do not already provide this) and conversation untangling [5]. That is, there may be no good segmentation of the conversation into contiguous temporal units, and we may instead need to untangle parallel threads in order to produce meaningful segments for indexing. This is particularly true when the goal is not merely retrieving a timestamp index into the audio/video, but when the user is interested in reading an automatic transcript of that portion of the conversation. Previous work in rich transcription has already shown that users can more quickly digest and apply information from a transcript than from a recording of the original conversation, and enriching the transcript beyond recognizer output further improves comprehension speed and accuracy [7]. Conversation untangling and segmentation would naturally extend this line of work. Of course, in some cases the natural unit of conversation may be particularly short, or the entire unit needed at retrieval time, such that no automatic segmentation is needed [23].

In comparison to WER and topic segmentation, back-channels and disfluency have garnered less attention [1]. This is not too surprising in that their impact on existing retrieval practices is likely less pronounced. Back-channels are simply words like "yeah", "uh-huh", etc. used to indicate to other participants in the conversation that a person listening is still actively engaged in that conversation. Disfluency consists of terms which disrupt the flow of conversation like filled pauses (e.g.. "uh", "um", "ah", etc.), explicit editing terms (e.g. "I mean", "that is", "I meant to say", etc.), and speech repairs (e.g. "I want a flight to Boston, no, Chicago") and restarts ("the dog is... where's my hat?") in which the literal transcript includes words which the speaker introduced into discourse by accident and abandoned. In regard to existing retrieval techniques, back-channels and disfluency can lead to two types of errors. Since different speakers use back-channels and disfluency with widely different rates, TF and segment length statistics can be significantly different for two segments relevant to the same topic. Secondly, the presence of repaired (accidental) terms such as "Boston" effectively adds to WER. Thinking beyond today's IR practices, "normalizing" the speech by recognizing and filtering out such terms has been shown to be useful in NLP analysis as well as rich transcription [14], and so may be similarly important in applying NLP to CS retrieval. While filled pauses are unambiguous and easily detected, recognizing back channels and other forms of disfluency is more difficult since the same words can be used to convey meaningful information [19].

Sentential structure is also vastly different between BN and CS, particularly as speakers trail off, interrupt one another, and revise their utterances on-the-fly. In a pure bag-of-words approach to retrieval, sentence boundaries are ignored, but use of phrases in retrieval typically improves precision around 10% [3]. Phrase-based statistics can be expected to perform best when not collected or applied across sentential boundaries, especially as phrase length increases, suggesting the potential importance of boundary detection. This issue has largely been ignored in retrieving text with a reasonable justification: sentences tend to be rather long in text (maybe around 30 words in a typical newspaper), and so error introduced for short phrase statistics by approximating the entire document as a single sentence is somewhat limited. However, sentence-like units (SUs) in conversational speech are distributed and behave much differently than their analogs in text (so much so that linguists have coined this new term to distinguish them) [14]. The most striking of these difference is with regard to retrieval is that SUs tend to be far shorter in length, perhaps around 6-7 words on average in telephone conversations, meaning that as segment size increases, assuming an entire CS segment represents a single SU will be an increasingly poor approximation. Thus, ignoring SU boundaries introduces a tension between longer segments better compensating for word recognition errors but further compromising any phrasal statistics collected.

My initial work in retrieving CS has focused on interview data used in the Cross-Language Speech Retrieval track at CLEF [17]. Queries come from real user requests, and the collection is pre-segmented and includes 1-best recognizer output, manual summary, manual keywords following a manually defined ontology, and some additional information [28]. While retrieval on this dataset addresses a practical need and provides an opportunity for multi-site comparison, it is somewhat limiting in terms of studying the various phenomena mentioned above: there is no manual transcription, nor have sentence boundaries, syntax, or disfluency been marked. For this reason, I am considering applying retrieval next to Fisher telephone conversations, for which speech recognizer lattices and n-best lists are available in addition to all of the manual data mentioned above [14]. The challenge in using this dataset is the lack of explicit queries and relevance assessments; it would be necessary to infer these from the pre-specified topics the participants discussed, which would certainly introduce a degree of artificiality into any results obtained [28]. However, I do not know of a better data set available for studying the phenomena above in order to assess the impact of each on retrieval accuracy (with a manual control condition), so this may be an acceptable trade-off. Using the Fisher data would also build on and complement previous work in rich transcription and syntactic analysis of CS [14].

5. EVALUATION

In this section, I briefly consider a few evaluation conditions and their potential impact on NLP-based retrieval.

TREC-style topics have been used for many years in TREC evaluations and express an information need at three levels of detail: *title*, *description*, and *narrative*. Most official TREC evaluations have focused on *title+description* queries, whereas published research tends to focus on *title*-only queries as being more indicative of the what is used to search the web today. I think there is also a less obvious reason prior work has focused on short queries: calculating document relevance in terms of TF-IDF style statistics, longer queries do not offer much additional information beyond slightly improved estimates, so there is little reason to justify the added burden on users when a short query does roughly as well. In contrast to this, people understand much more from longer queries. For example, in reading TREC topic #2, a person can learn the following: the information need has something to do with “acquisitions” (*title*), the acquisition should be currently proposed and must involve a US company and a foreign one (*description*), and that both companies must be identified by name (*narrative*). The point to make here is that if the query is specified so poorly that a person cannot understand it any better than a TF-IDF system (i.e. *title* query), then greater use of NLP has little hope of improving system precision. However, as query length increases, the gap in comprehension grows significantly between a TF-IDF system and a person, and thus there is a potential benefit to be derived from deeper analysis. The current focus on short, keyword queries may also be creating something of a perpetuating cycle: users write short keyword queries based on their experience using current retrieval systems, and TR research focuses on such queries because that is what users tend to write. If we were to instead adopt a longer-term perspective, it is easy to imagine users writing more complete, expressive queries if systems could justify this user effort with significantly improved precision (just think of the natural conversations people have with their neighborhood librarians in describing what they are looking for). Previous work in applying NLP to retrieval has already shown improvement over existing practice with increasing query length [3], but presumably the present gap between practices is not sufficiently large to justify the additional processing time required to apply those NLP techniques. I plan to evaluate my own experiments on the various query lengths for comparison purposes, but I will focus on longer queries where the NLP's potential is clearer.

With regard to tasks and evaluation metrics, I have focused thus far on ad hoc TR scored by mean average position (MAP) following the dominant trend in the literature. However, it may be that NLP has more to offer for a variant task or usage metric. Brants hypothesized filtering, passage-based retrieval, or a metric that considers only a small set of the top documents retrieved would better suit NLP because the system would have fewer opportunities to deliver relevant information [3]. Sparck Jones has made a similar point that NLP may be most beneficial in identifying relevant information within its larger context [35]. This question of appropriate tasks and evaluation metrics when using NLP will be another dimension of consideration in my work.

6. CONCLUSION

This paper introduced my dissertation study exploring methods for integrating modern NLP with state-of-the-art IR techniques. After briefly surveying current practice in text retrieval, I discussed conversational speech data and the unique set of considerations it presents with regard to retrieval. With retrieval precision of both forms of language typically below 50%, the time seems ripe to once more look beyond the existing class of TF-IDF statistics to NLP and see what leverage its modern practice might have to offer.

7. ACKNOWLEDGMENTS

This work was initiated at the Institute of Formal and Applied Linguistics (ÚFAL) at Charles University in Prague, which graciously hosted the author for the first half of 2007. Support was provided by NSF PIRE Grant No OISE-0530118 and DARPA GALE contract HR0011-06-2-0001. Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the supporting agencies.

8. REFERENCES

[1] J. Allan. Perspectives on information retrieval and speech. In *Information Retrieval Techniques for Speech Applications: LNCS 2273*, pages 1–10, 2002.

[2] J. Bai, D. Song, P. Bruza, J.-Y. Nie, and G. Cao. Query expansion using term relationships in language models for information retrieval. In *Proceedings of the 14th conference on Information and knowledge management (CIKM)*, pages 688–695, 2005.

[3] T. Brants. Natural Language Processing in Information Retrieval. In *Proceedings of the 14th Meeting of Computational Linguistics in the Netherlands*, 2003.

[4] G. Cao, J.-Y. Nie, and J. Bai. Integrating word relationships into language models. In *Proc. of the 28th SIGIR conference on Research and development in information retrieval*, pages 298–305, 2005.

[5] S. A. Çamtepe, M. K. Goldberg, M. Magdon-Ismail, and M. Krishn. Detecting conversing groups of chatters: a model, algorithms, and tests. In *Proceedings of the IADIS International Conference on Applied Computing*, pages 89–96, 2005.

[6] E. Charniak. Immediate-head parsing for language models. In *Proc. of the Assoc. for Computational Linguistics (ACL)*, pages 116–123, 2001.

[7] D. Jones et al. Measuring the readability of automatic speech-to-text transcripts. In *Proc. of Eurospeech*, pages 1585–1588, 2003.

[8] O. Etzioni, M. Banko, and M. J. Cafarella. Machine reading. In *Proceedings of the 21th National Conference on Artificial Intelligence (AAAI)*, 2006.

[9] H. Fang, T. Tao, and C. Zhai. A formal study of information retrieval heuristics. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 49–56, 2004.

[10] J. Gao, J.-Y. Nie, G. Wu, and G. Cao. Dependence language model for information retrieval. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 170–177, 2004.

[11] J. Gao, H. Qi, X. Xia, and J.-Y. Nie. Linear discriminant model for information retrieval. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 290–297, 2005.

[12] J. Garofolo, G. Auzanne, and E. Voorhees. The trec spoken document retrieval track: A success story. In *the Ninth Text Retrieval Conference (TREC-9)*, 1999.

[13] J. T. Goodman. A bit of progress in language modeling extended version. Technical Report 2001-72, Microsoft Research, 2001.

[14] M. Harper, B. Dorr, J. Hale, B. Roark, I. Shafran, M. Lease, Y. Liu, M. Snover, L. Yung, A. Krasnyanskaya, and R. Stewart. *2005 Johns Hopkins Summer Workshop Final Report on Parsing and Spoken Structural Event Detection*.

[15] B. Hui. Applying NLP to IR: Why and how. Technical report, Department of Computer Science, University of Waterloo, April 1998.

[16] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proceedings of the 24th ACM SIGIR conference*, pages 120–127, 2001.

[17] M. Lease and E. Charniak. Brown at CL-SR'07: Retrieving conversational speech in English and Czech. In *Proceedings of the Cross-Language Evaluation Forum (CLEF): Cross-Language Speech Retrieval (CL-SR) track*, 2007.

[18] M. Lease, E. Charniak, M. Johnson, and D. McClosky. A look at parsing and its applications. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence (AAAI-06)*, 16–20 July 2006.

[19] M. Lease, M. Johnson, and E. Charniak. Recognizing disfluencies in conversational speech. *IEEE Transactions on Audio, Speech and Language Processing*, 14(5):1566–1573, September 2006.

[20] C. Lee, G. G. Lee, and M. G. Jang. Dependency structure applied to language modeling for information retrieval. *ETRI Journal*, 28:337–346, 2006.

[21] D. D. Lewis and K. Sparck Jones. Natural language processing for information retrieval. *Communications of the ACM*, 39(1):92–101, 1996.

[22] M. Marcus et al. Building a large annotated corpus of English: The Penn Treebank. *Comp. Linguistics*, 19(2):313–330, 1993.

[23] J. Mamou, D. Carmel, and R. Hoory. Spoken document retrieval from call-center conversations. In *29th SIGIR conference on research and development in information retrieval*, pages 51–58, 2006.

[24] C. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[25] D. McClosky, E. Charniak, and M. Johnson. Effective self-training for parsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 152–159, 2006.

[26] D. Metzler and W. B. Croft. Linear feature-based models for information retrieval. *Information Retrieval*, 10(3):257–274, 2007.

[27] R. Nallapati and J. Allan. Capturing term dependencies using a language model based on sentence trees. In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 383–390, 2002.

[28] D. Oard et al. Overview of the CLEF-2006 cross-language speech retrieval track. In *Working Notes for the Cross Language Evaluation Forum 2006 Workshop*, 2006.

[29] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference*, pages 275–281, 1998.

[30] S. Robertson. The probability ranking principle in IR. *Journal of Documentation*, 33(4):294–304, 1977.

[31] M. Saraclar and R. Sproat. Lattice-based search for spoken utterance retrieval. In *HLT-NAACL 2004: Main Proceedings*, pages 129–136, 2004.

[32] A. Singhal. Modern information retrieval: A brief overview. *Bulletin IEEE Computer Society Technical Committee on Data Engineering*, 24:35–43, 2001.

[33] A. F. Smeaton. Using NLP or NLP resources for information retrieval tasks. In T. Strzalkowski, editor, *Natural language information retrieval*, pages 99–111. Kluwer Academic Publishers, Dordrecht, NL, 1999.

[34] F. Song and W. B. Croft. A general language model for information retrieval. In *Proceedings of the eighth international conference on Information and knowledge management (CIKM)*, pages 316–321, 1999.

[35] K. Sparck Jones. What is the role of NLP in text retrieval? In T. Strzalkowski, editor, *Natural language information retrieval*, pages 1–21. Kluwer Academic Publishers, Dordrecht, NL, 1997.

[36] K. Sparck Jones, S. Walker, and S. Robertson. A probabilistic model of information retrieval: development and comparative experiments (parts i and ii). *Information Processing and Management*, 36:779–840, 2000.

[37] C. J. Van Rijsbergen. *Information Retrieval, 2nd edition*. Dept. of Computer Science, University of Glasgow, 1979.

[38] C. Zhai. A brief review of information retrieval models. Technical report, Department of Computer Science, University of Illinois at Urbana-Champaign, 2007.

[39] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst.*, 22(2):179–214, 2004.

[40] L. Zhou and D. Zhang. NLPIR: A theoretical framework for applying natural language processing to information retrieval. *Journal of the American Society for Information Science and Technology*, 54(2):115–123, 2003.