

SEMANTIC DATA SELECTION FOR VERTICAL BUSINESS VOICE SEARCH

Giuseppe Di Fabbrizio, Diamantino Caseiro, Amanda J. Stent

AT&T Labs - Research, Inc.
Florham Park, NJ 07932 - USA

{pino,dcaseiro,stent}@research.att.com

ABSTRACT

Local business voice search is a popular application for mobile phones, where hands-free interaction and speed are critical to users. However, speech recognition accuracy is still not satisfactory when the number of businesses and locations is extended nationwide. For mobile users, searching a local business directory is often related to the fulfillment of specific tasks “on-the-move”, such as finding a restaurant, a movie theater, or a retailer chain. Restricting the local search to specific domains improves the quality of search results. In this paper, we present a new approach to data selection for bootstrapping and optimizing language models for vertical business sectors by exploiting semantic knowledge encoded in the business database and in the business category taxonomy. We demonstrate that, in the case of queries in the restaurant domain and without collecting new data, speech recognition word accuracy improves by 9.5% relative when compared with a generic local business language model.

Index Terms— Local business search, voice search, language modeling.

1 Introduction

Speech recognition has been recently applied as a hands-free input modality to mobile devices for local business search. In the last three years, several web-based search companies added speech input capabilities to their mobile search applications¹, aiming to attract more traffic for their advertising campaigns. More specifically, local business voice search (LBVS) [1, 2, 3], is becoming one of the most popular applications among mobile users, since it better addresses the search precision needs of so-called *road warriors*, who must often deal with limited device interface capabilities since they are away from their desktop-based search engines.

In most current mobile LBVS applications, the automatic speech recognizer (ASR) uses one large language model, typically built from mixed data sources, including web search logs, business database entries, and transcriptions of utterances from real users. User queries may include either the name of a business or a particular business category, optionally followed by a specific location. Natural language

¹e.g., m.bing.com, m.google.com, m.yelp.com

queries tend also to include additional carrier phrases (e.g., *find me ...*, *where is the ...*, *I'm looking for ...*, etc.) that do not provide extra information for the search task, and additional constraints (e.g., *CVS pharmacies open 24 hours*, *Italian restaurants with music*, etc.) based on specific features of the business. Although a broad language model gives the ASR the flexibility to accept any combination of spoken terms in a single utterance, subsequent processing stages only extract information relevant to the search engine, ignoring any extra constraints.

In [4] for instance, a spoken query like *Ton's Mongolian Grill in Arlington, Texas*, is first recognized by the ASR, then is further processed to extract the search term (*Ton's Mongolian Grill*) and the location term (*Arlington, Texas*); other words are discarded. Finally, the query terms are passed to the search engine, which returns matching business listings to the mobile device.

However, mobile users of LBVSs are often focused on fulfilling a specific task “on-the-move” such as finding a restaurant, a movie theater, or a retail chain. Accordingly to [5, chap. 9], the top five queries from the `YELLOWPAGES.COM` logs include: 1) Restaurants, 2) Movie theaters, 3) Pizza, 4) Walmart, and 5) Animal shelters. This suggests that a local search for the restaurant domain could represent a significant amount of mobile user search traffic and might be an interesting area to customize voice search. In fact, several text-based restaurant search applications are already migrating from the desktop to mobile devices: `HAVE2EAT` [6], `Zagat`, `UrbanSpoon`, `Yelp`, and `YP Mobile`² (none of these are currently speech-enabled with the exception of `YP Mobile`).

In this paper, we present a new approach for bootstrapping and optimizing language models for the restaurant domain, starting from a generic local business search application. Our method has several advantages: it provides a fast way to create new language models from existing generic query data; it generalizes across domains; and it is compatible with existing query parsing techniques. In experiments conducted using test queries sampled from an existing LBVS system, we achieved a speech recognizer word accuracy improvement of

²Respectively: `www.have2eat.com`, `mobile.zagat.com`, `www.urbanspoon.com`, `m.yelp.com`, `m.yelp.com`.

9.5% relative when compared to a general local business language model, and a two-fold speech recognition speedup using the same working point as the generic voice search model.

The rest of this paper is organized as follows: Section 2 describes the overall system architecture. In Section 3, we present our procedure for creating the language modeling training corpus. Section 4 illustrates how to create the five reference models we used in our experiment. The experimental setup is described in Section 5, the conclusions are described in Section 6, and the future work plan in Section 7.

2 System description

The system architecture of the best performing configuration is illustrated in Figure 1. The main data feed is generated by the transcribed utterances collected by SPEAK4IT³ [7], a LBVS system running on the Apple iPhone and Android-based mobile phones which has been deployed since December 2008. It retrieves business listings from the YELLOWPAGES.COM search engine and displays them on an interactive map.

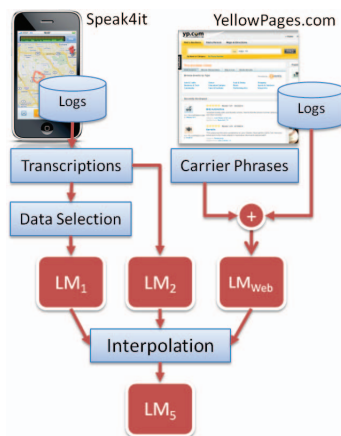


Fig. 1: System architecture

The *data selection* module, described in detail in Section 3, selects queries concerning the restaurant domain. These queries are used to create LM_1 , a domain-dependent language model, while the full set of transcribed utterances is used to create LM_2 . Separately, query log feeds from the YELLOWPAGES.COM text-based, web-based local business search are augmented with carrier phrases, such as “BUSINESS in LOCATION”, and used to create a domain-independent LM_{web} model.

From these component language models, we build several interpolated language models: LM_3 , a weighted interpolation of LM_1 and LM_2 ; LM_4 , a weighted interpolation of LM_2 and LM_{web} , and LM_5 , a weighted interpolation of LM_1 , LM_2 , and LM_{web} which contains contributions from all the various data sources.

³www.speak4it.com

3 Corpus construction

Our domain-specific corpus was built by selecting restaurants and food-related queries from the SPEAK4IT utterance transcriptions. To select queries related to the restaurant domain we adopt the three-step approach illustrated in Figure 2.

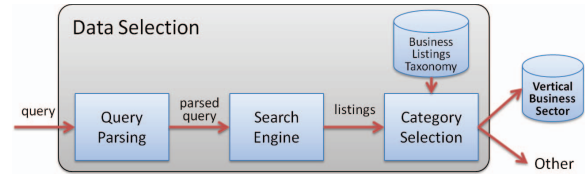


Fig. 2: Vertical business sector data selection process

First, we parse SPEAK4IT transcribed queries using the query parser described in [8]. Query text is separated into three fields: *SearchTerm*, *LocationTerm*, and *Filler*. Because over 80% of SPEAK4IT voice queries do not include location terms, search queries are extended, when available, with the location of the user as provided by the phone’s GPS. If the location is missing, the query is extended nationwide. In the second step, the *Fillers* are discarded, and the remaining fields are used to submit a query to the YELLOWPAGES.COM search engine. For search engines that already support single field query input, query parsing can be skipped.

3.1 Taxonomy-based data selection

The business category selection module assigns a business vertical to the query using the top two business categories assigned to the query by the search engine. We used the YELLOWPAGES.COM search engine, but any local business search engine can be used as long as the categories are consistent with the terms used in the following modules. Product and service taxonomies are commercially available, minable from open directory web sites such as dmz.org, or available as open source collaborative efforts such as freebase.com.

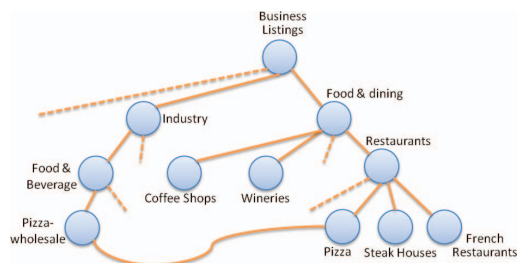


Fig. 3: Business listing taxonomy fragment

For each category, the business category selection module traverses a business taxonomy graph (Figure 3) and assigns the overall business category by using the category of the ancestor node one level below the taxonomy root. The taxonomy graph expresses *is-a* relationships among local business

Data	Words	Utterances
SPEAK4IT Queries	889,661	355,679
Selected Queries	252,978	112,840
Test set	3,937	1,697
Dev set	3,980	1,687

Table 1: Restaurant data statistics

search categories and, in our case, can go up to six nodes deep with categories getting more specialized going down and away from the root. Cycles in the graph always include the root node. For the experiments described here, if any of the selected parent nodes are labeled with a restaurant related category, the query is assigned to the restaurant business vertical. For instance, the query *Olive Garden in Madison New Jersey* is associated with two category nodes: *Italian Restaurants* and *Take Out Restaurants*, both children of the node *Restaurants* which is in turn a child of the more general category *Food & Dining*. In our case, if a query belongs to multiple-matching categories, it is still a valid restaurant query if any of its categories belongs to the *Food & Dining* category.

3.2 Data selection evaluation

The statistics of the selected corpus are presented in Table 1. Restaurant queries make up around 32% of the original corpus, confirming the relevance of this vertical sector for mobile users. The separation between testing, development and training utterances was done chronologically. The transcribed utterances from the last and second-last weekly data sets were used for testing and development respectively. All the other previously transcribed sets were assigned to the training data.

To evaluate the quality of the data selection process, we randomly selected 216 utterances and manually verified the accuracy of the classification. For each entry, we checked if the query included a name for a business or a category valid for a restaurant domain search query. For instance, when searching nationwide, the query *Barriques* returned both listings for wine bars and restaurants, but the restaurants appeared at the end of the listings, so the categories from the taxonomy did not match the restaurant domain. Similarly, *Bill's Cafe* can be either a bar or a restaurant. A few other misclassifications included wrong locations - *White Castle in Windsor, Ontario, Canada*. Overall, only 5 queries were misclassified, giving a classification accuracy of around 98%.

4 Models

Using the available training data, five different language models were built. LM_1 is a 3-gram LM built from the selected SPEAK4IT query transcriptions. LM_2 is a 3-gram LM built from all SPEAK4IT utterance transcriptions. LM_3 is the linear interpolation of LM_1 and LM_2 . The interpolation weights, 0.8 and 0.2 respectively, were optimized on the development data. LM_4 is a general purpose language model for business search. This model is the linear interpolation of two models: LM_2 and a 3-gram language model built from 100 million YELLOWPAGES.COM web queries (LM_{web}).

Model	Word acc.	Sentence acc.
LM_1	76.3	66.3
LM_2	76.7	66.7
LM_3	77.8	67.4
LM_4	78.9	69.7
LM_5	80.9	71.8

Table 2: Accuracy of different language models

The web queries consist of two separate fields, business name (BN) and location (LOC). These fields were embedded in carrier phrases to extend the coverage of spoken queries in the corpus. For each web query, a number of queries were generated by embedding business name and location in carrier phrase templates. The counts of the sentences thus generated were weighted by the relative frequency of the carrier phrase templates observed in the SPEAK4IT voice queries.

LM_5 is the interpolation of LM_1 with the component language models of LM_4 . The interpolation weights were also optimized on the development set and are shown in equation 1:

$$P_{LM_5}(w|h) = 0.184P_{web}(w|h) + 0.074P_{LM_2}(w|h) + 0.742P_{LM_1}(w|h) \quad (1)$$

Notice that the in-domain model LM_1 is the major contributor to the interpolated model. All models are Katz's backoff n-gram language models trained using *grmtools* [9], which have support for the fractional counts weighting the carrier phrases in the web language model.

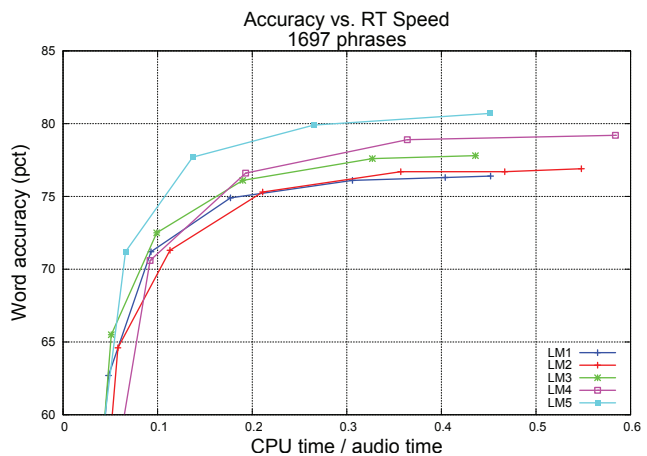


Fig. 4: Accuracy vs. real time factor of different language models

5 Experiments

The AT&T Watson [10] recognizer was used to evaluate the word accuracy of the language models described in Section 4. All experiments used a triphonic HMM acoustic model originally developed for SPEAK4IT.

Table 2 shows the performance of the various models. It is clear that using all available data improves accuracy relative to using only the in-domain data subset. One important reason is the added coverage brought by using queries for businesses unrelated to restaurants. The names of some of these businesses can partially overlap with the names of restaurants. This happens often with business names that refer to common or local landmarks or to family names. Also, non-restaurant queries may include carrier phrases not included in our carrier phrase patterns or in the in-domain data.

Interpolation achieves a good balance between the specificity of in-domain data and the added coverage of the larger data sets; LM_3 achieves the best performance of the models trained on transcriptions only, and LM_5 improves on the general search LM_4 by strongly weighting the contribution of the in-domain data.

The more accurate models are also faster, as we can see in Figure 4. In particular, LM_5 is about 2x faster than LM_4 .

To measure the impact of these models on search accuracy, one of the authors labeled the recognized queries in the test data for type of ASR error. ‘Not serious’ ASR errors include number errors (e.g., *restaurants* vs. *restaurant*), presence of function words (e.g., *in*, *at*, *the*) or disfluencies (e.g., *ah*), spelling errors not likely to affect search results (e.g., *alehouse* vs. *ale house*), and extra words not likely to affect search results (e.g., *krispy kreme donuts* vs. *krispy kreme*). ‘Serious’ ASR errors include all other errors in names, categories or location information. Rates of ‘serious’ ASR error were 16.67%, 17.3% and 14.77% for LM_1 , LM_2 and LM_3 for category queries, and 25.92%, 26.25% and 23.71% for LM_1 , LM_2 and LM_3 for business name queries. These results show that weighting language models towards popular business verticals can lead to improvements in search accuracy equal to or greater than the corresponding improvements in ASR accuracy.

6 Conclusions

This paper describes a process for specializing a general voice search language model to a specific vertical business sector such as restaurant search. The process is based on selecting a relevant subset of user queries by submitting them to the search engine and selecting those with at least one relevant business in the top n results. Business relevance is determined using a business listing taxonomy. The selected data is used to build statistical language models which are combined with general business search language models. This combination enables the model to specialize to the vertical business sector without losing coverage. In the restaurant domain, this process improves word accuracy by 9.5% relative when compared to a general business search language model, and produces fewer ASR errors likely to impact search.

7 Future work

The accuracy obtained with the best model described in this paper is sufficient to support a speech version of a mobile ap-

plication such as HAVE2EAT⁴. Once this version is deployed, we will collect and transcribe user queries. These will be used, at first, as development and test data for tuning the language model, and later, once sufficient data is collected, to train an additional model to be integrated in equation 1. We also plan to use *click logs* from the existing HAVE2EAT mobile application. These can be converted into text and used in a similar way as the web logs. A more ambitious future goal is to expand the functionality of the application by allowing search based on features extracted from restaurant reviews.

References

- [1] A. Acero, N. Bernstein, R. Chambers, Y. C. Jui, X. Li, J. Odell, P. Nguyen, O. Scholz, and G. Zweig, “Live search for mobile: Web services by voice on the cellphone,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008.
- [2] S. Chang, S. Boyce, K. Hayati, I. Alphonso, and B. Buntschuh, “Modalities and demographics in voice search: learnings from three case studies,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008.
- [3] M. Bacchiani, F. Beaufays, J. Schalkwyk, M. Schuster, and B. Strope, “Deploying goog-411: early lessons in data, measurement, and testing,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008.
- [4] J. Feng and S. Bangalore, “Effects of word confusion networks on voice search,” in *EACL ’09: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, Morristown, NJ, USA, 2009, pp. 238–245, Association for Computational Linguistics.
- [5] T. Holloway, “The big picture: Search and discovery,” in *Beautiful Visualization*, J. Steelea and N. Iliinskyn, Eds., chapter 9. O’Reilly Media, Inc., 2010.
- [6] G. Di Fabrizio, N. Gupta, S. Besana, and P. Mani, “Have2eat: A Restaurant Finder with Review Summarization for Mobile Phones,” in *The 23rd International Conference on Computational Linguistics (COLING 2010)*, Beijing, China, August 23-27 2010.
- [7] J. Feng, S. Bangalore, and M. Gilbert, “Role of natural language understanding in voice local search,” in *10th Annual Conference of the International Speech Communication Association (Interspeech 2009)*, 2009.
- [8] J. Feng and S. Bangalore, “Query parsing for voice-enabled mobile local search,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2009.
- [9] C. Allauzen, M. Mohri, and B. Roark, “A general weighted grammar library,” in *Implementation and Application of Automata, 9th International Conference, CIAA 2004*, Kingston, Canada, July 22-24 2004, pp. 23–34.
- [10] V. Goffin, C. Allauzen, E. Bocchieri, D. Hakkani-Tür, A. Ljolje, S. Parthasarathy, M. Rahim, G. Riccardi, and M. Saracilar, “The AT&T WATSON Speech Recognizer,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2005.

⁴www.have2eat.com