Populating semantic classes using large-scale corpora

Laura Stoia, Tianfang Xu, Donna K. Byron and **Eric Fosler-Lussier**
Speech and Language Technologies Lab
Department of Computer Science and Engineering
The Ohio State University
2015 Neil Ave, 395 Dreese Labs
Columbus, Ohio USA
stoia, xut, dbyron, fosler@cse.ohio-state.edu

Abstract

Our research developing a system to provide natural language driving-directions has given us the need to infer a database of landmarks. Automatically bootstrapping the list of landmarks for a particular city is the goal of the system described in this paper. The method finds word sequences that appear in contexts similar to a set of seed items, then uses web page counts, provided by the Google API, to estimate word frequencies, and finally selects candidate terms using mutual information. Human evaluators judged an average of 77 % of the items produced to be useful landmarks.

1 Introduction

To collaborate with a human user on a task, spoken dialog systems rely on semantic classes to partition the world into objects of different types having different properties. For example, a dialog system that supplies the user with driving directions will need to distinguish between cities, major highways, local roads, and destination addresses, to name just a few important classes. Developing an inventory of semantic classes and populating those categories with instances is one of the most time-consuming aspects of spoken dialog system development.

Systems built to work in the real world, such as the Communicator systems (Walker et al., 2001) or MIT's Galaxy/Voyager (Glass et al., 1995; Goddeau et al., 1994), face several challenges. The world is dynamically changing and the lists of instances in each semantic class must be kept up to date. Also, the system may need to populate the instances in each category for a particular locality, such as a list of restaurants in a certain town. Inducing set members for some semantic classes is simply a matter of using existing resources. For example, a list of cities in the US or airports could be easily found in existing data resources.

However, other classes that might be useful for a particular system may be harder to define, and may have no existing database to draw from. In our case, our work is motivated by the need to develop semantic and lexical resources for a spoken dialog system for driving directions. Useful, naturalistic driving directions include not only names of streets and the distance between turning points, for which data is readily available[1] but also landmarks such as a road that one passes just before a turn, or a conspicuous building located at the turning point. Populating the list of landmarks for a particular city is the goal of the system described in this paper.

The special challenge in this task is that determining which items can be landmarks is an ill-defined concept, yet we would like to collect a set of instances of this concept. The approach adopted in this paper is to let the concept develop organically based on the language usage found in a large corpus, namely the World Wide Web. Our semi-automatic method starts from a set of seed terms and finds items mentioned in similar contexts, then calculates mutual information to determine which items should be kept. An important feature of this specific method is that there is no reliance on language resources such as part-of-speech taggers or parsing. Therefore, the method can be utilized for languages other than English, even those for which lexical and grammatical resources do not exist. The method requires only small amounts of human intervention in the initial step and in filtering the final results and can be employed for semi-automatic extraction of instances of a particular semantic class from the web. Automatic bootstrapping techniques such as this provide a vital reduction of the time spent developing knowledge resources for spoken dialog systems.

[1] http://www.census.gov/geo/www/tiger

(a) **Raw sequences w/o nonwords**	**(b) Best k-gram from each sequence**			
	L^*	Point-wise mutual info $M(L^*)$	final weight $I(L^*)$	#times best
Nationwide Arena Administrative Address	Nationwide Arena	0.35	0.00092	1
by appointment Ohio Craft Museum	Craft Museum	-1.02	-0.00025	3
Polaris Amphitheater Columbus	Polaris Amphitheater	3.97	0.00097	1
appointment top Ohio Craft Museum	Craft Museum	-1.02	-0.00025	3
Shopping HOTEL Hyatt Regency Columbus	Hyatt Regency	3.78	0.03483	1
top SHOPPING Columbus City Center	City Center	-2.59	-0.05732	3
garden COSI Columbus Web Site	COSI Columbus	-1.18	-0.00077	4
around them Elijah Pierce Gallery	Elijah Pierce Gallery	0.76	0.00001	1
art and culture Heritage Museum	Heritage Museum	-1.05	-0.00046	1
Columbus City Center Web Site	City Center	-2.59	-0.05732	3
Connie Golden COSI Columbus	COSI Columbus	-1.18	-0.00077	4
availability Holiday Inn City Center	Holiday Inn	2.38	0.18024	1
spot Franklin Park Conservator	Franklin Park	-1.88	-0.00810	1
Exhibition COSI Columbus	COSI Columbus	-1.18	-0.00077	4
Map it COSI Museum	COSI Museum	0.19	0.00002	1
it Polaris Fashion Place	Polaris Fashion Place	2.72	0.00254	2
official property websites COSI Columbus	COSI Columbus	-1.18	-0.00077	4
floats on the Scioto River	Scioto River	-0.07	-0.00010	1
standing homes in Franklin County	Franklin County	-0.66	-0.01022	1
closed Monday Olentangy Indian Caverns	Olentangy Indian Caverns	6.43	0.00153	1
and services Columbus City Center	City Center	-2.67	-0.05732	3
for shopping Polaris Fashion Place	Polaris Fashion Place	2.64	0.00254	2
sculpture livestock competitions and concerts	competitions and	-1.80	-0.00124	1
Report Welcome to Cleveland Columbus	Welcome to	-1.04	-0.04486	1

Table 1: Isolating informative k-grams from raw sequences

2 Background and Related Work

2.1 Why use landmarks?

Research in navigation has shown that people prefer directions that include landmarks (May et al., 2003; Johnston et al., 2002; Burnett, 2000). There has been much recent interest in development of spoken dialog systems for travel and tourism domains(Glass et al., 1995; Goddeau et al., 1994; Hansen et al., 2000; Dale et al., forthcoming), and these systems often include directions. It has been noted that including items of interest or landmarks would be beneficial to these systems(Johnston et al., 2002).

2.2 What is a landmark?

Landmarks are those object in the environment that aide the user in navigating and understanding the space (Brenner and Elias, 2003). Using a list of businesses is not a viable solution, because we do not know how salient they are to a particular conversation: are they big enough to be well known, or are they visible from the street? Commercial systems that involve direction giving rely on a hand-made database of points of interest that can become obsolete with time.[2] (Sorrows and Hirtle, 1999; Brenner and Elias, 2003) argued that landmarks can be divided into three categories: visual, cognitive and structural, and an appropriate choice of landmarks depends on factors such as the navigation context or application mode: is the user a pedestrian or driver? Every place name could be a potential landmark, but this depends on the real navigation task. Therefore, collecting a list of landmarks appropriate for a particular navigation task is best accomplished by starting from a list of known items, and finding additional items used in similar contexts.

3 Inducing Landmarks from the Web

The World Wide web is a free resource of language data, and it contains relatively up-to-date information. Researchers have used the web to construct knowledge bases for particular domains (Craven et al., 2000; Ferguson et al., 2002). One particularly useful resource for this task is the Google API,[3] which has been used to bootstrap related terms and form a corpus from the web (Baroni and Bernardini, 2004; Sato and Sasaki, 2003). The Google API provides an automated interface to the Google

[2] http://www.navteq.com

[3] http://www.google.com/apis

search engine, which returns both links to web pages containing the query term as well as an estimated count of the number of documents containing the term. Therefore, the API can be used both for estimating the prior probability of lexical strings and for finding examples of the contexts in which the strings are used. Our technique harnesses the power of the web to find items that are used as landmarks, and exploits the observation that multiple landmarks within a city are likely to appear in the same document.

N-gram obtain from "available Holiday Inn City Center"	Count	P-wise M
availability	113000	-2.74
Holiday	89800	-2.97
Inn	240000	-1.99
City	1280000	-0.31
Center	730000	-0.88
availability Holiday	29	-5.29
Holiday Inn	**133000**	**2.38**
Inn City	4530	-3.65
City Center	39800	-2.59
availability Holiday Inn	13	-4.10
Holiday Inn City	3060	-1.06
Inn City Center	3960	-2.90
availability Holiday Inn City	1	-6.34
Holiday Inn City Center	2610	-0.34
availability Holiday Inn City Center	1	-5.46

Table 2: Initial filtering

The algorithm starts by searching a small set of known landmarks to generate a corpus of documents that contain numerous landmarks in addition to the original seeds. Next, word sequences that are likely to contain landmarks are extracted from that corpus, and mutual information is used to delimit meaningful collocations from each sequence. Finally, the list of proposed landmarks is sorted and a cutoff point is set, below which items are discarded.

3.1 Building a corpus of web pages

The process for building the corpus begins with choosing a target city and a short list of known landmarks to initiate the web search. We choose Columbus, OH as our target city and distributed the seeds between various types of landmarks – a zoo, a theater, a river,a boat, an official building and an art center.[4] We then queried the web using the Google API, which returned a list of html documents matching each seed (e.g., "Columbus Zoo" + Columbus + OH). For each seed, the html text for the 180 highest-ranking matches was obtained, resulting in an 8000k word corpus of html text.

3.2 Extracting tentative landmarks

One criteria that must be met by an item to function as a landmark is that it must name an object that exists at a physical location. The first five words that come before a sequence resembling an address were treated as a tentative landmark. For our purposes, we considered an address to be a word sequence containing at least one digit (street number) followed by no more than 10 words (street name) followed by "Columbus, OH" (or "Columbus, Ohio"). We chose to limit the window for a proposed landmark to $n = 5$ words because allowed us to limit the number of Google queries yet provided sufficient coverage since most landmark names are ≤ 5 words. We removed all non-alphabetic characters from the original 5-word window, resulting in proposed items with length $n \leq 5$ (see Table 1 (a) for examples).

3.3 Initial term filtering

We used a measure similar to mutual information to isolate meaningful subsequences from the tentative landmark names proposed in the preceding step. The probability of a k-word sequence ($1 \leq k \leq 5$) was calculated relative to the number of documents returned for a query on Columbus+OH ($\approx 1,760,000$). The Google search engine was used to obtain document counts.

$$P(w_i...w_{i+k-1}) = \frac{count(\text{“}w_i...w_{i+k-1}\text{“}+Columbus+OH)}{count(Columbus+OH)}$$

The point-wise mutual information for all contiguous k-word subsequences of the tentative landmark proposed by the previous stage was computed using an adaptation[5] of the formula in (Church and Hanks, 1989):

$$M(w_i...w_{i+k-1}) = log\frac{P(w_i.....w_{i+k-1})}{P(w_i)....P(w_{i+k-1})}$$

Table 2 shows the result of calculating this on an example sequence. The point-wise mutual information gives a measure of how likely words are to co-occur in sequence compared to the probability that they occur independently.

[4]The specific seeds were: Columbus Zoo, COSI, Ohio Theater, Santa Maria, Scioto River, State Capitol and Wexner Center.

[5]This measure is similar to point-wise mutual information, but the counts are obtained from the number of hits returned by Google, not from the number of occurrences in the corpus formed by all web pages for Columbus,Oh. We assumed that the relative proportions and rankings are still valid.

This measure has been used in similar situations for semantic clustering in limited domains (Pargellis et al., 2004).

Because point-wise mutual information does not give reliable results for small values, we removed all sequences with a hit count of 15 or less. To discount single words with very big counts, since these turn out to be usually just common words, we eliminated single words with hit counts greater than 150,000. There are also groups of common words that are very frequent not because they are good landmarks, but because they are a good collocation of common words. For this reason we excluded sequences with a hit count greater than 50,000. These constants were chosen by the experimenters by examining the number of hits obtained on our seeds. More research in finding the optimal cut-off values automatically is planned. See Table 2 for an example of this filtering. The items "city", "center", "Inn" were eliminated because they have counts $> 150,000$; "availability Holiday Inn City", "availability Holiday Inn City Center", "availability Holiday Inn" were eliminated because they have counts < 15. The final ordered list is: "Holiday Inn" 2.3851, "Holiday Inn City Center" -0.3473, "City Center" -2.5907, "availability" -2.7456, "Holiday Inn City" -2.9059, "Inn City Center" -2.9059, "Inn City" -3.6514, "availability Holiday" -5.2923.

At the end of this stage for each tentative landmark we returned the subsequence from each raw sequence with the largest value of M that satisfied the filtering constraints above:

$$L^* = argmax_{i,k} M(w_i ... w_{i+k-1})$$

3.4 Final term filtering

The final stage of the process was to choose which instances to propose as candidates for actual landmarks. Under the assumption that good landmarks should be items that are commonly known, they should be mentioned often on the web. We kept only the proposed items that were the best k-gram from at least 2 raw word sequences, using the procedure described in the previous section. The list was reordered based on the mutual information $I(L^*)$ contained in the words

$$I(L^*) = P(L^*) * M(L^*)$$

Proposed landmarks
Holiday Inn
Holiday Inn Express
Santa Maria
Vice President
Crowne Plaza
Hyatt Regency
Blue Jackets
Courtyard by Marriott
Embassy Suites
Hall of Fame
Executive director
Comfort Suites
Symphony Orchestra
Comfort Inn
Schiller Park
Hawthorn Suites
Baymont Inn
Port Columbus International Airport
Ohio State University
Performing Arts

Table 3: Best 20 landmarks found by our method

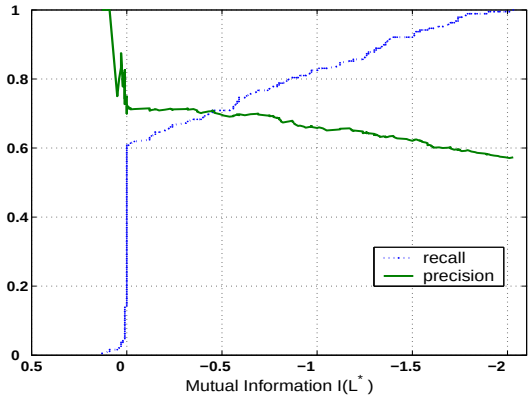

Figure 1: Precision and recall trade-off

4 Results

Our final list contains 320 proposed landmarks. The top 20 results are shown in Table 3. As the table shows, the list includes many names of places that were found on web pages alongside the seeds, such as hotels, the airport and our university.

Evaluating instances of landmarks is very difficult, since there is some disagreement in the navigation literature (Sorrows and Hirtle, 1999). The evaluation of the landmarks proposed by our system has two parts. First we measure its performance in proposing meaningful noun-phrases, and next human evaluators were presented with a list of items to judge whether they can be used as landmarks.

4.1 Selection of noun phrases

For the first part of the evaluation two judges labelled each landmark as either a valid noun-phrase or incomplete or word salad sequences (one was the lead researcher and one was a linguistic student at OSU). The inter-coder reliability coefficient (Carletta, 1996) for this task was $kappa = 0.93$. The percentage of complete noun phrases proposed by our algorithm was 85.6%, very high considering we do not use any grammatical resources or POS tags and we have no previous knowledge of the vocabulary.

4.2 Selection of landmarks

For the evaluation of our landmark extraction results we removed all items not referring to locations (such as "Vice President" and "Executive director" in Table 3) and used 4 naive evaluators (computer science graduate students outside our research group) to judge our results. They were presented with a list of 225 proposed landmarks. The judges were asked to imagine a scenario where they are giving directions to a mall to a person familiar with the area and to judge whether each item would match the template sentence "just after you go past item/the item, the mall is on your right", and label each item with yes/no/not sure. The result of this part of the study showed that deciding upon using an item in giving directions is not a straightforward task. The average pairwise kappa was 0.57, but people considered on the average that 77% of the items they were evaluating were useful as landmarks (the non-location items removed from the evaluation by the experimenters were labelled as "no". Factoring in this number, 54% out of the 320 items retained after the final stage of our algorithm were labelled as 'yes'). Mutual information proved to be a good evaluation criteria, as there was high correlation between a high $I(L^*)$ score and precision. We calculated the recall out of all the landmarks found. Fig 1 shows the tradeoff between recall and precision across different cutoff values of $I(L^*)$ for the items evaluated in our experiment.

4.3 Discussion of Results

Because it finds some invalid word sequences, the method described in this paper is proposed as a semi-automatic method of discovering the list of landmarks for a particular locality. The list of proposed landmarks must be vetted by a human judge before they could be put to use in a direction-giving system. One might assume that a similar list of landmarks could be generated through a completely manual process simply by developing a list of expected landmark types, such as hotels, stadiums, and museums. However, finding the instances that match each category requires some work and familiarity with the town, for example "Polaris Fashion Place" is a shopping mall in Columbus, although its name does not reveal this fact. In addition, our method has an advantage over manual acquisition because it discovers many individual landmarks that would be unlikely to be discovered if one were working from a predetermined list of landmark types. For example, in Columbus, the "Field of Corn" (a public statuary installation) is a useful landmark, although this word sequence would appear to be a non-landmark to those not familiar with the area. Also included in our induced list are named regions such as the "Brewery District" and "The Short North", corporate headquarters when they are conspicuous, such as the Longaberger Baskets headquarters, and entertainment venues such as a go-kart/miniature golf facility. For the amount of human time required to seed the procedure and look over the final results, the coverage obtained is superior to totally manual methods.

The method does not depend on any linguistic resources such as POS taggers or grammars, or even morphological indicators of named entities such as capitalization. It only needs a small set of seeds and knowledge about a loose format of addresses in that particular country, which are straightforward to obtain. The Google API allows the calling program to restrict the results of the queries to specific languages. The term filtering process that our method utilizes, relying on mutual information to judge the informativeness of word sequences, is assumed to work across languages.

In this initial study we used our intuition about landmarks existing at a physical location to design an extraction criteria, but there are other possible patterns to induce the first raw sequences (like landmark...number Miles). Future research is planned to automatically learn these patterns and classifying the types of landmarks according to the context in which they appear.

By using different seed items and a clue other than proximity to an address in the selection of initial targets, this method should be useful for

populating other sorts of semantic classes for different language domains. The method can also easily be utilized for different languages.

5 Conclusions

This paper describes a semi-automatic method to discover members of a semantic class using World Wide Web pages as a corpus. The method finds word sequences that appear in contexts similar to a set of seed items, then uses web page counts, provided by the Google API, to estimate word frequencies, and finally selects candidate terms using mutual information. This method is particularly useful for discovering members of categories such as landmarks, which are defined by usage rather than by an analytic definition, and for which no gazetteers are available. The method is language-independent and can be used to bootstrap information about any locality of interest.

6 Acknowledgments

This work was funded by the Ohio State University Department of Computer Science and Engineering. The authors wish to thank the anonymous reviewers for their helpful comments and suggestions.

References

Marco Baroni and Silvia Bernardini. 2004. Bootcat: bootstrapping corpora and terms from the web. In *Proceedings of LREC 2004*.

C. Brenner and B. Elias. 2003. Extracting landmarks for car navigation systems using existing gis databases and laser scanning. ISPRS Archives, Vol. 34, sep.

Gary Burnett. 2000. Turn right at the traffic lights: The requirement for landmarks in vehicle navigation systems. *The Journal of Navigation*, 53:499–510.

Jean Carletta. 1996. Assessing agreement on classification tasks: the kappa statistic. *Computational Liguistics*, 22(2):249–254.

Kenneth Ward Church and Patrick Hanks. 1989. Word association norms, mutual information, and lexicography. In *Proceedings of the 27th conference on Association for Computational Linguistics*, pages 76–83. Association for Computational Linguistics.

Mark Craven, Dan DiPasquo, Dayne Freitag, Andrew K. McCallum, Tom M. Mitchell, Kamal Nigam, and Sean Slattery. 2000. Learning to construct knowledge bases from the world wide web. *Aritificial Intelligence*, 118:69–113.

Robert Dale, Sabine Gelfof, and Jean-Philippe Proust. forthcoming. Using natural language generation in automatic route description. *Journal of Research and Practice in Information Technology*.

George Ferguson, James Allen, Nate Blaylock, Donna Byron, Nate Chambers, Myroslava Dzikovska, Lucian Galescu, Xipeng Shen, Robert Swier, and Mary Swift. 2002. The Medication Advisor Project: Preliminary report. Technical Report TR776, University of Rochester Computer Science Department.

J. Glass, G. Flammia, D. Goodine, M. Phillips, J. Polifroni, S. Sakai, S. Seneff, and V. Zue. 1995. Multilingual spoken-language understanding in the MIT Voyager system. *Speech Communication*, 17(1-2):1–18, August.

D. Goddeau, E. Brill, J. Glass, C. Pao, M. Phillips, J. Polifroni, S. Seneff, and V. Zue. 1994. Galaxy: A human-language interface to on-line travel information. In *Proceedings of the International Conference on Spoken Language Processing*, pages 707–710, Yokohama, Japan, September.

John H. L. Hansen, Jay Plucienkowski, Stephen Gallant, Bryan Pellom, and Wayne Ward. 2000. "cu-move": Robust speech processing for in-vehicle speech systems. *International Conference on Spoken Language Processing -ICSLP*.

M. Johnston, S. Bangalore, G. Vasireddy, A. Stent, P. Ehlen, M. Walker, S. Whittaker, and P. Maloor. 2002. Match: An architecture for multimodal dialogue systems. In *Proceedings of ACL 2002*, pages 376–383.

Andrew J. May, Tracy Ross, and Steven H. Bayer. 2003. Drivers' information requirements when navigating in an urban environment. *The Journal of Navigation*, 56:89–100.

A. Pargellis, E. Fosler-Lussier, C.-H. Lee, A. Potamianos, and A. Tsai. 2004. Automatic induction of semantic classes for spoken dialogue systems. *Speech Communication*, 43(3):188–203.

Satoshi Sato and Yasuhiro Sasaki. 2003. Automatic collection of related terms from the web. In *ACL-03 Companion Volume to the Proceedings of the Conference*.

Molly Sorrows and Stephen Hirtle. 1999. The nature of landmarks for real and electronic spaces. In *Proceedings of the International Conference on Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science*, volume 1661, pages 37–50. Springer Verlag.

Marilyn A. Walker, Rebecca Passonneau, and Julie E. Boland. 2001. Quantitative and qualitative evaluation of darpa communicator spoken dialogue systems. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-01)*.