Finding Parts in Very Large Corpora

Matthew Berland, Eugene Charniak
mb, ec@cs.brown.edu
Department of Computer Science
Brown University, Box 1910
Providence, RI 02912

Abstract

We present a method for extracting parts of objects from wholes (e.g. "speedometer" from "car"). Given a very large corpus our method finds part words with 55% accuracy for the top 50 words as ranked by the system. The part list could be scanned by an end-user and added to an existing ontology (such as WordNet), or used as a part of a rough semantic lexicon.

1 Introduction

We present a method of extracting parts of objects from wholes (e.g. "speedometer" from "car"). To be more precise, given a single word denoting some entity that has recognizable parts, the system finds and rank-orders other words that may denote parts of the entity in question. Thus the relation found is strictly speaking between words, a relation Miller [7] calls "meronymy." In this paper we use the more colloquial "part-of" terminology.

We produce words with 55% accuracy for the top 50 words ranked by the system, given a very large corpus. Lacking an objective definition of the part-of relation, we use the majority judgment of five human subjects to decide which proposed parts are correct. The program's output could be scanned by an end-user and added to an existing ontology (e.g., WordNet), or used as a part of a rough semantic lexicon.

To the best of our knowledge, there is no published work on automatically finding parts from unlabeled corpora. Casting our nets wider, the work most similar to what we present here is that by Hearst [5] on acquisition of hyponyms ("isa" relations). In that paper Hearst (a) finds lexical correlates to the hyponym relations by looking in text for cases where known hyponyms appear in proximity (e.g., in the construction (NP, NP and (NP other NN)) as in "boats, cars, and other vehicles"), (b) tests the proposed patterns for validity, and (c) uses them to extract relations from a corpus. In this paper we apply much the same methodology to the part-of relation. Indeed, in [5] Hearst states that she tried to apply this strategy to the part-of relation, but failed. We comment later on the differences in our approach that we believe were most important to our comparative success.

Looking more widely still, there is an ever-growing literature on the use of statistical/corpus-based techniques in the automatic acquisition of lexical-semantic knowledge ([3,4,6,8,9,10]). We take it as axiomatic that such knowledge is tremendously useful in a wide variety of tasks, from lower-level tasks like noun-phrase reference, and parsing to user-level tasks such as web searches, question answering, and digesting. Certainly the large number of projects that use WordNet [7] would support this contention. And although WordNet is hand-built, there is general agreement that corpus-based methods have an advantage in the relative completeness of their coverage, particularly when used as supplements to the more labor-intensive methods.

2 Finding Parts

2.1 Parts

Webster's Dictionary defines "part" as "one of the often indefinite or unequal subdivisions into which something is or is regarded as divided and which together constitute the whole." The vagueness of this definition translates into a lack of guidance on exactly what constitutes a part, which in turn translates into some doubts about evaluating the results of any procedure that claims to find them. More specifically, note that the definition does not claim that parts must be physical objects. Thus, say, "novel" might have "plot" as a part.

In this study we handle this problem by asking informants which words in a list are parts of some target word, and then declaring majority opinion to be correct. We give more details on this aspect of the study later. Here we simply note that while our subjects often disagreed, there was fair consensus that what might count as a part depends on the nature of the word: a physical object yields physical parts, an institution yields its members, and a concept yields its characteristics and processes. In other words, "floor" is part of "building" and "plot" is part of "book."

2.2 Patterns

Our first goal is to find lexical patterns that tend to indicate part-whole relations. Following Hearst [5], we find possible patterns by taking two words that are in a part-whole relation (e.g, basement and building) and finding sentences in our corpus (we used the North American News Corpus (NANC) from LDC) that have these words within close proximity. The first few such sentences are:

... the **basement** of the **building**.
... the **basement** in question is in a four-story apartment **building** ...
... the **basement** of the apartment **building**.
From the **building**'s **basement** ...
... the **basement** of a **building** ...
... the **basement**s of **building**s ...

From these examples we construct the five patterns shown in Table 1. We assume here that parts and wholes are represented by individual lexical items (more specifically, as head nouns of noun-phrases) as opposed to complete noun phrases, or as a sequence of "important" noun modifiers together with the head. This occasionally causes problems, e.g., "conditioner" was marked by our informants as not part of "car", whereas "air conditioner" probably would have made it into a part list. Nevertheless, in most cases head nouns have worked quite well on their own.

We evaluated these patterns by observing how they performed in an experiment on a single example. Table 2 shows the 20 highest ranked part words (with the seed word "car") for each of the patterns A-E. (We discuss later how the rankings were obtained.)

Table 2 shows patterns A and B clearly outperform patterns C, D, and E. Although parts occur in all five patterns, the lists for A and B are predominately parts-oriented. The relatively poor performance of patterns C and E was anticipated, as many things occur "in" cars (or buildings, etc.) other than their parts. Pattern D is not so obviously bad as it differs from the plural case of pattern B only in the lack of the determiner "the" or "a". However, this difference proves critical in that pattern D tends to pick up "counting" nouns such as "truckload." On the basis of this experiment we decided to proceed using only patterns A and B from Table 1.

A. *whole* NN[-PL] 's POS *part* NN[-PL]
   . . . building's basement . . .

B. *part* NN[-PL] of PREP {the|a} DET *mods* [JJ|NN]* *whole* NN
   . . . basement of a building . . .

C. *part* NN in PREP {the|a} DET *mods* [JJ|NN]* *whole* NN
   . . . basement in a building . . .

D. *parts* NN-PL of PREP *wholes* NN-PL
   . . . basements of buildings . . .

E. *parts* NN-PL in PREP *wholes* NN-PL
   . . . basements in buildings . . .

Format: type_of_word TAG type_of_word TAG ...
NN = Noun, NN-PL = Plural Noun
DET = Determiner, PREP = Preposition
POS = Possessive, JJ = Adjective

Table 1: Patterns for partOf(basement,building)

3 Algorithm

3.1 Input

We use the LDC North American News Corpus (NANC). which is a compilation of the wire output of several US newspapers. The total corpus is about 100,000,000 words. We ran our program on the whole data set, which takes roughly four hours on our network. The bulk of that time (around 90%) is spent tagging the corpus.

As is typical in this sort of work, we assume that our evidence (occurrences of patterns A and B) is independently and identically distributed (iid). We have found this assumption reasonable, but its breakdown has led to a few errors. In particular, a drawback of the NANC is the occurrence of repeated articles; since the corpus consists of all of the articles that come over the wire, some days include multiple, updated versions of the same story, containing identical paragraphs or sentences. We wrote programs to weed out such cases, but ultimately found them of little use. First, "update" articles still have substantial variation, so there is a continuum between these and articles that are simply on the same topic. Second, our data is so sparse that any such repeats are very unlikely to manifest themselves as repeated examples of part-type patterns. Nevertheless since two or three occurrences of a word can make it rank highly, our results have a few anomalies that stem from failure of the iid assumption (e.g., quite appro-

Pattern A
headlight windshield ignition shifter dashboard radiator brake tailpipe pipe airbag speedometer converter hood trunk visor vent wheel occupant engine tyre
Pattern B
trunk wheel driver hood occupant seat bumper backseat dashboard jalopy fender rear roof windshield back clunker window shipment reenactment axle
Pattern C
passenger gunmen leaflet hop houseplant airbag gun koran cocaine getaway motorist phone men indecency person ride woman detonator kid key
Pattern D
import caravan make dozen carcass shipment hundred thousand sale export model truckload queue million boatload inventory hood registration trunk ten
Pattern E
airbag packet switch gem amateur device handgun passenger fire smuggler phone tag driver weapon meal compartment croatian defect refugee delay

Table 2: Grammatical Pattern Comparison

priately, "clunker").

Our seeds are one word (such as "car") and its plural. We do not claim that all single words would fare as well as our seeds, as we picked highly probable words for our corpus (such as "building" and "hospital") that we thought would have parts that might also be mentioned therein. With enough text, one could probably get reasonable results with any noun that met these criteria.

3.2 Statistical Methods

The program has three phases. The first identifies and records all occurrences of patterns A and B in our corpus. The second filters out all words ending with "ing", "ness", or "ity", since these suffixes typically occur in words that denote a quality rather than a physical object. Finally we order the possible parts by the likelihood that they are true parts according to some appropriate metric.

We took some care in the selection of this metric. At an intuitive level the metric should be something like $p(w \mid p)$. (Here and in what follows w denotes the outcome of the random variable generating wholes, and p the outcome for parts. $W(w)$ states that w appears in the patterns AB as a whole, while $P(p)$ states that p appears as a part.) Metrics of the form $p(w \mid p)$ have the desirable property that they are invariant over p with radically different base frequencies, and for this reason have been widely used in corpus-based lexical semantic research [2,9,10]. However, in making this intuitive idea someone more precise we found two closely related versions:

$$p(w, W(w) \mid p)$$
$$p(w, W(w) \mid p, P(p))$$

We call metrics based on the first of these "loosely conditioned" and those based on the second "strongly conditioned".

While invariance with respect to frequency is generally a good property, such invariant metrics can lead to bad results when used with sparse data. In particular, if a part word p has occurred only once in the data in the AB patterns, then perforce $p(w \mid p)$ = 1 for the entity w with which it is paired. Thus this metric must be tempered to take into account the quantity of data that supports its conclusion. To put this another way, we want to pick (w, p) pairs that have two properties, $p(w \mid p)$ is high and $\mid w, p \mid$ is large. We need a metric that combines these two desiderata in a natural way.

We tried two such metrics. The first is Dunning's [1] log-likelihood metric which measures how "surprised" one would be to observe the data counts $\mid w, p \mid, \mid \neg w, p \mid, \mid w, \neg p \mid$ and $\mid \neg w, \neg p \mid$ if one assumes that $p(w \mid p) = p(w)$. Intuitively this will be high when the observed $p(w \mid p) \gg p(w)$ and when the counts supporting this calculation are large.

The second metric is proposed by Johnson (personal communication). He suggests asking the question: how far apart can we be sure the distributions $p(w \mid p)$and $p(w)$ are if we require a particular p level, say .05 or .01. We call this new test the "significant-difference" test, or sigdiff. Johnson observes that compared to sigdiff, log-likelihood tends to overestimate the importance of data frequency at the expense of the distance between $p(w \mid p)$ and $p(w)$.

3.3 Comparison

Table 3 shows the 20 highest ranked words for each statistical method, using the seed word "car." The first group contains the words found for the method we perceive as the most accurate, sigdiff and strong conditioning. The other groups show the differences between them and the first group. The + category means that this method adds the word to its list, − means the opposite. For example, "back" is on the sigdiff-loose list but not the sigdiff-strong list.

In general, sigdiff worked better than surprise and strong conditioning worked better than loose condi-

Sigdiff, Strong	
	airbag brake bumper dashboard driver fender headlight hood ignition occupant pipe radiator seat shifter speedometer tailpipe trunk vent wheel windshield
Sigdiff, Loose	
+	back backseat oversteer rear roof vehicle visor
−	airbag brake bumper pipe speedometer tailpipe vent
Surprise, Strong	
+	back cost engine owner price rear roof use value window
−	airbag bumper fender ignition pipe radiator shifter speedometer tailpipe vent
Surprise, Loose	
+	back cost engine front owner price rear roof side value version window
−	airbag brake bumper dashboard fender ignition pipe radiator shifter speedometer tailpipe vent

Table 3: Methods Comparison

tioning. In both cases the less favored methods tend to promote words that are less specific ("back" over "airbag", "use" over "radiator"). Furthermore, the combination of sigdiff and strong conditioning worked better than either by itself. Thus all results in this paper, unless explicitly noted otherwise, were gathered using sigdiff and strong conditioning combined.

4 Results

4.1 Testing Humans

We tested five subjects (all of whom were unaware of our goals) for their concept of a "part." We asked them to rate sets of 100 words, of which 50 were in our final results set. Tables 6 - 11 show the top 50 words for each of our six seed words along with the number of subjects who marked the word as a part of the seed concept. The score of individual words vary greatly but there was relative consensus on most words. We put an asterisk next to words that the majority subjects marked as correct. Lacking a formal definition of part, we can only define those words as correct and the rest as wrong. While the scoring is admittedly not perfect[1], it provides an adequate reference result.

	book	**building**	**car**
10	8	7	8
20	14	12	17
30	20	18	23
40	24	21	26
50	28	29	31
	hospital	**plant**	**school**
10	7	5	10
20	16	10	14
30	21	15	20
40	23	20	26
50	26	22	31

Table 4: Result Scores

Table 4 summarizes these results. There we show the number of correct part words in the top 10, 20, 30, 40, and 50 parts for each seed (e.g., for "book", 8 of the top 10 are parts, and 14 of the top 20). Overall, about 55% of the top 50 words for each seed are parts, and about 70% of the top 20 for each seed. The reader should also note that we tried one ambiguous word, "plant" to see what would happen. Our program finds parts corresponding to both senses, though given the nature of our text, the industrial use is more common. Our subjects marked both kinds of parts as correct, but even so, this produced the weakest part list of the six words we tried.

4.2 WordNet

WordNet	
+	door engine floorboard gear grille horn mirror roof tailfin window
−	brake bumper dashboard driver headlight ignition occupant pipe radiator seat shifter speedometer tailpipe vent wheel windshield

Table 5: WordNet Comparison

We also compared out parts list to those of WordNet. Table 5 shows the parts of "car" in WordNet that are not in our top 20 (+) and the words in our top 20 that are not in WordNet (−). There are definite tradeoffs, although we would argue that our top-20 set is both more specific and more comprehensive. Two notable words our top 20 lack are "engine" and "door", both of which occur before 100. More generally, all WordNet parts occur somewhere before 500, with the exception of "tailfin", which never occurs with car. It would seem that our program would be excellent tool for expanding Wordnet, as a person can scan and mark the list of part words in a few minutes.

[1]For instance, "shifter" is undeniably part of a car, while "production" is only arguably part of a plant.

5 Discussion and Conclusions

The program presented here can find parts of objects given a word denoting the whole object and a large corpus of unmarked text. The program is about 55% accurate for the top 50 proposed parts for each of six examples upon which we tested it. There does not seem to be a single cause for the 45% of the cases that are mistakes. We present here a few problems that have caught our attention.

Idiomatic phrases like "a jalopy of a car" or "the son of a gun" provide problems that are not easily weeded out. Depending on the data, these phrases can be as prevalent as the legitimate parts.

In some cases problems arose because of tagger mistakes. For example, "re-enactment" would be found as part of a "car" using pattern B in the phrase "the re-enactment of the car crash" if "crash" is tagged as a verb.

The program had some tendency to find qualities of objects. For example, "driveability" is strongly correlated with car. We try to weed out most of the qualities by removing words with the suffixes "ness", "ing", and "ity."

The most persistent problem is sparse data, which is the source of most of the noise. More data would almost certainly allow us to produce better lists, both because the statistics we are currently collecting would be more accurate, but also because larger numbers would allow us to find other reliable indicators. For example, idiomatic phrases might be recognized as such. So we see "jalopy of a car" (two times) but not, of course, "the car's jalopy". Words that appear in only one of the two patterns are suspect, but to use this rule we need sufficient counts on the good words to be sure we have a representative sample. At 100 million words, the NANC is not exactly small, but we were able to process it in about four hours with the machines at our disposal, so still larger corpora would not be out of the question.

Finally, as noted above, Hearst [5] tried to find parts in corpora but did not achieve good results. She does not say what procedures were used, but assuming that the work closely paralleled her work on hyponyms, we suspect that our relative success was due to our very large corpus and the use of more refined statistical measures for ranking the output.

6 Acknowledgments

This research was funded in part by NSF Grant NSF grant IRI-9319516, ONR Grant N0014-96-1-0549, and the Brown University UTRA program. Thanks to the entire statistical NLP group at Brown, and particularly to Mark Johnson, Brian Roark, Gideon Mann, and Ana-Maria Popescu who provided invaluable help on the project.

References

1. Dunning, T. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics 191* (1993), 61–74.
2. Gale, W. A., Church, K. W. and Yarowsky, D. A method for disambiguating word senses in a large corpus. *Computers and the Humanities* (1992).
3. Grefenstette, G. SEXTANT: extracting semantics from raw text implementation details. *Heuristics: The Journal of Knowledge Engineering* (1993).
4. Hatzivassiloglou, V. and McKeown, K. R. *Predicting the semantic orientation of adjectives.* In *Proceedings of the 35th Annual Meeting of the ACL*. 1997, 174–181.
5. Hearst, M. *Automatic acquisition of hyponyms from large text corpora.* In *Proceedings of the Fourteenth International Conference on Computational Linguistics,*. 1992.
6. Lin, D. *Automatic retrieval and clustering of similar words.* In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*. 1998, 768–774.
7. Miller, G., Beckwith, R., Fellbaum, C., Gross, D. and Miller, K. J. WordNet: an online lexical database. *International Journal of Lexicography 3* (1990), 235–245.
8. Richardson, S. D., Dolan, W. B. and Vanderwende, L. *MindNet: acquiring and structuring semantic information from text.* In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*. 1998, 1098–1102.
9. Riloff, E. and Shepherd, J. *A corpus-based approach for building semantic lexicons.* In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*. 1997, 117–124.
10. Roark, B. and Charniak, E. *Noun-phrase co-occurrence statistics for semi-automatic semantic lexicon construction.* In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*. 1998, 1110–1116.

Ocr.	Frame	Word	x/5
853	3069	author	5*
23	48	subtitle	4*
114	414	co-author	4*
7	16	foreword	5*
123	963	publication	2
5	10	epigraph	3*
9	32	co-editor	4*
51	499	cover	5*
220	3053	copy	2
125	1961	page	5*
103	1607	title	5*
6	28	authorship	2
13	122	manuscript	2
45	771	chapter	5*
4	14	epilogue	5*
69	1693	publisher	4*
16	240	jacket	5*
48	1243	subject	5*
2	2	double-page	0
289	10800	sale	0
12	175	excerpt	2
45	1512	content	5*
16	366	plot	5*
3	10	galley	2
57	2312	edition	3*
8	123	protagonist	4*
3	13	co-publisher	3*
6	82	spine	5*
13	360	premise	1
11	295	revelation	2
30	1390	theme	2
3	16	fallacy	2
53	3304	editor	5*
9	252	translation	2
44	2908	character	5*
23	1207	tone	2
8	218	flaw	2
56	4265	section	4*
15	697	introduction	5*
47	3674	release	1
2	5	diarist	0
3	22	preface	4*
6	140	narrator	4*
8	276	format	2
3	25	facsimile	0
3	26	mock-up	1
5	111	essay	2
35	3648	back	5*
6	194	heroine	4*
7	300	pleasure	0

Table 6: book

Ocr.	Frame	Word	x/5
72	154	rubble	0
527	2116	floor	5*
42	156	facade	4*
85	456	basement	5*
100	577	roof	5*
9	23	atrium	4*
32	162	exterior	5*
28	152	tenant	1
12	45	rooftop	4*
49	333	wreckage	1
7	20	stairwell	5*
30	250	shell	0
14	89	demolition	0
14	93	balcony	5*
10	60	hallway	5*
23	225	renovation	0
4	9	janitor	1
10	62	rotunda	5*
36	432	entrance	3*
7	37	hulk	0
82	1449	wall	5*
23	276	ruin	0
37	572	lobby	5*
12	120	courtyard	4*
3	6	tenancy	0
13	156	debris	1
9	83	pipe	2
32	635	interior	3*
219	6612	front	4*
7	58	elevator	5*
11	143	evacuation	1
2	2	web-site	0
2	2	airshaft	4*
2	2	cornice	3*
47	1404	construction	2
9	115	landlord	1
14	285	occupant	1
129	5616	owner	1
17	404	rear	3*
25	730	destruction	1
15	358	superintendent	1
3	11	stairway	5*
6	72	cellar	5*
3	12	half-mile	0
37	1520	step	5*
10	207	corridor	5*
39	1646	window	5*
2	3	subbasement	5*
38	1736	door	4*
4	31	spire	3*

Table 7: building

Ocr.	Frame	Word	x/5
92	215	trunk	4*
27	71	windshield	5*
12	24	dashboard	5*
13	30	headlight	5*
70	318	wheel	5*
9	21	ignition	4*
43	210	hood	5*
119	880	driver	1
6	13	radiator	5*
4	6	shifter	1
37	285	occupant	1
15	83	brake	5*
5	12	vent	3*
6	18	fender	5*
3	4	tailpipe	5*
8	42	bumper	5*
11	83	pipe	3*
7	36	airbag	5*
108	1985	seat	4*
3	5	speedometer	4*
3	6	converter	2
3	6	backseat	5*
64	1646	window	5*
28	577	roof	5*
2	2	jalopy	0
33	784	engine	5*
20	404	rear	4*
4	19	visor	3*
6	68	deficiency	0
75	3648	back	2
2	3	oversteer	1
10	216	plate	3*
9	179	cigarette	1
3	13	clunker	0
7	117	battery	5*
18	635	interior	3*
19	761	speed	1
11	334	shipment	0
5	73	re-enactment	0
3	18	conditioner	2
3	18	axle	5*
11	376	tank	5*
6	125	attribute	0
18	980	location	1
71	6326	cost	1
5	88	paint	4*
4	51	antenna	5*
2	5	socket	0
2	5	corsa	0
6	151	tire	5*

Table 8: car

Ocr.	Frame	Word	x/5
43	302	ward	5*
3	7	radiologist	5*
2	2	trograncic	0
3	9	mortuary	4*
3	9	hopewell	0
17	434	clinic	5*
3	11	aneasthetist	5*
18	711	ground	1
16	692	patient	4*
33	2116	floor	4*
68	5404	unit	4*
44	3352	room	2
11	432	entrance	4*
19	1237	doctor	5*
15	1041	administrator	5*
6	207	corridor	4*
25	2905	staff	3*
35	5015	department	5*
7	374	bed	5*
2	11	pharmacist	4*
100	23692	director	5*
5	358	superintendent	3*
3	89	storage	3*
20	5347	chief	2
4	299	lawn	2
4	306	compound	0
29	13944	head	0
3	149	nurse	5*
2	33	switchboard	4*
3	156	debris	0
14	5073	executive	2
2	35	pediatrician	4*
17	7147	board	1
13	4686	area	1
4	416	ceo	2
5	745	yard	2
15	6612	front	3*
8	2200	reputation	1
3	190	inmate	1
4	457	procedure	2
2	42	overhead	0
14	6315	committee	4*
5	875	mile	0
15	7643	center	1
2	46	pharmacy	4*
4	518	laboratory	5*
16	8788	program	1
2	48	shah	0
29	25606	president	2
3	276	ruin	1

Table 9: hospital

Ocr.	Frame	Word	x/5
185	1404	construction	2
5	12	stalk	4*
23	311	reactor	3*
8	72	emission	3*
10	122	modernization	1
2	2	melter	3*
19	459	shutdown	1
6	62	start-up	0
41	1663	worker	2
22	844	root	3*
17	645	closure	0
22	965	completion	0
26	1257	operator	4*
12	387	inspection	2
21	980	location	2
19	856	gate	3*
2	4	sprout	3*
4	41	leaf	5*
26	1519	output	2
3	20	turbine	3*
12	506	equipment	3*
4	51	residue	1
2	5	zen	0
3	22	foliage	4*
8	253	conversion	0
8	254	workforce	1
8	309	seed	3*
17	1177	design	4*
9	413	fruit	5*
23	1966	expansion	2
5	131	pollution	2
50	6326	cost	1
24	2553	tour	0
24	2564	employee	5*
29	3478	site	1
40	5616	owner	3*
9	577	roof	4*
49	7793	manager	3*
41	6360	operation	3*
6	276	characteristic	1
21	2688	production	3*
3	48	shoot	0
32	5404	unit	1
6	337	tower	1
5	233	co-owner	1
2	13	instrumentation	3*
8	711	ground	2
3	69	fiancee	0
5	296	economics	1
7	632	energy	2

Table 10: plant

Ocr.	Frame	Word	x/5
525	1051	dean	4*
164	445	principal	5*
134	538	graduate	3*
11	24	prom	3*
7	12	headmistress	4*
16	61	alumni	3*
19	79	curriculum	5*
4	5	seventh-grader	3*
8	22	gymnasium	5*
25	134	faculty	5*
3	3	crit	0
13	87	endowment	3*
8	40	alumnus	2
9	57	cadet	0
11	82	enrollment	2
5	18	infirmary	4*
3	5	valedictorian	4*
8	52	commandant	0
75	1462	student	5*
56	1022	feet	0
10	100	auditorium	5*
4	15	jamieson	0
5	26	yearbook	3*
8	71	cafeteria	4*
28	603	teacher	5*
4	17	grader	2
2	2	wennberg	0
2	2	jeffe	0
7	65	pupil	3*
21	525	campus	4*
11	203	class	5*
17	423	trustee	3*
8	115	counselor	4*
7	108	benefactor	2
5	56	berth	0
5	60	hallway	4*
7	130	mascot	3*
39	2323	founder	1
2	4	raskin	0
6	112	playground	4*
105	8788	program	3*
16	711	ground	3*
6	120	courtyard	3*
25	1442	hall	4*
17	837	championship	1
3	20	accreditation	2
6	135	fellow	1
2	5	freund	0
4	53	rector	2
6	144	classroom	4*

Table 11: school