Analogy as an Organizational Principle in the Construction of Large Knowledge-Bases

Tony Veale, Guofu Li

Abstract A capacity for analogy is an excellent acid test for the quality of a knowledge-base. A good knowledge-base should be balanced and coherent, so that its high-level generalities are systematically reflected in a variety of lower-level specializations. As such, we can expect a rich, well-structured knowledge-base to support a greater diversity of analogies than one that is imbalanced, disjoint or impoverished. We argue here that the converse is also true: when choosing from a large pool of candidate propositions, in which many propositions are invalid because they are extracted automatically from corpora or volunteered by untrained web-users, we should prefer those that are most likely to enhance the analogical productivity of the knowledge-base. We present a simple and efficient means of finding potential analogies within a large knowledge-base, using a corpus-constrained notion of pragmatic comparability rather than the typically less-constrained notion of semantic similarity. This allows us to empirically demonstrate, in the context of a substantial knowledge-base of simple generalizations automatically extracted from the Google n-grams, that knowledge acquisition proceeds at a significantly faster pace when candidate additions are prioritized according to their analogical potential.

1 Introduction

As a knowledge-base grows in scale and diversity, its capacity for insightful analogy should grow also ([19]). Analogy is a knowledge-hungry cognitive mechanism for learning and generalization, one that allows us to project our knowledge of one domain onto another ([9], [12]). Computational approaches to analogy work best when different domains are represented in largely isomorphic ways that facilitate systematic mappings, but this is a challenge for very large KBs constructed by diverse teams of engineers (e.g., [14]). Fortunately, the converse also seems to be

Tony Veale
Web Science and Technology Division, KAIST, Yuseong, Korea. e-mail: tony.veale@gmail.com

true: if we design a KB to maximize its capacity for analogy, the resulting structure should be well-balanced, coherent and rich in isomorphisms. So given a large pool of candidate propositions to choose from, we should prefer those that increase the potential for analogy in the growing KB.

A good knowledge-base is like a city park or a fashionable nightspot: each is subject to *neighborhood effects*, insofar as their appeal is in direct proportion to the value of the members they can attract ([29]). An empty nightclub has little appeal, but this grows as new high-profile members are added. Each new member provides what economists call a *positive externality* ([29]), since each addition makes a club or park livelier and more enjoyable for others. Likewise, each addition to a KB can have marginal benefits for the propositions that are already there, by increasing their connectedness and making it likelier that they can be used in longer and more complex inferential chains. But economists also speak of *negative externalities*, and invalid propositions can undermine the workings of a knowledge-base just as surely as drunken gatecrashers can detract from the appeal of a trendy nightspot. Nightclubs use doormen to discourage negative externalities while expediting the entry of those candidates queuing outside that are most aligned with the club's image. We similarly want to expedite the addition of candidate propositions that provide the most positive fit to the current knowledge-base, as measured in terms of their analogical potential. In effect, analogy is our doorman to the KB.

Though analogy is cognitively important ([9],[12],[7],[24]), it is just one of many capacities that we expect from our knowledge-bases, and many KB applications demand no analogical competence at all. So in this chapter we also consider a practical corollary, and ask whether incrementally constructing a knowledge-base so as to maximize its capacity for analogy results in a faster and more streamlined acquisition process overall. We have strong *a priori* reasons to presume so: balance, coherence and wide coverage are desirable qualities in any knowledge-base, while the value of high-level generalizations lies in the number and variety of lower-level specializations that they can explain. Analogy can directly pinpoint those additions that will flesh out the knowledge-base in the most coherent and well-rounded fashion.

Our argument is presented with an empirical evaluation in the following sections. We first discuss related background work on analogy and knowledge representation in section 2, before section 3 recasts analogy in terms of a corpus-based notion of comparability. Section 4 describes how a simple but relatively large knowledge-base of generic propositions is extracted from the Google n-grams ([3]). A coarse net is trawled over these n-grams, so many more invalid propositions are retrieved than are actually valid or desirable. Section 5 presents a model of analogy-guided acquisition, in which a mixed bag of candidate propositions is sorted according to their analogical *fit* to the current KB. Section 6 presents an evaluation of this approach, and quantifies the efficiency gains that arise from the use of analogy to shake out the best candidates. Final remarks are offered in section 7.

2 Related Work and Ideas

Computational treatments of analogy have traditionally focused on two principal forms. The first form, typified by the work of Gentner ([9]) and Holyoak and Thagard ([12]) and formally evaluated by Veale and Keane ([24]), is the *system analogy* that builds rich networks of cross-domain mappings via the systematic alignment of two complex structures. These analogies can be quite deep, and are most common in scientific reasoning, legal argumentation and education. The second form, typified by the work of Hofstadter ([11]) and Turney ([22], [23]), limits itself to the fixed frame A:B::C:D. These proportional analogies have, for many years, been a fixture of IQ and SAT-style aptitude tests ([25]). Despite a simplicity of form, these often highlight the subtle nuances that separate close conceptual neighbors, as in *mercenary:soldier::hack:writer*. Both forms have attracted very different computational treatments, yet each is mutually compatible, since complex system analogies can be seen as coherent global combinations of smaller, more local, proportional analogies.

System analogies are typically modeled using a structure-mapping process applied to graph representations of domain knowledge ([7]), in which mappings are constructed by finding the largest sub-graph isomorphisms between two logical representations. Finding optimal mappings is thus an NP-hard problem ([24]), so various heuristics and pragmatic assumptions are employed to generate good mappings in polynomial time. For Hofstadter and his fluid analogies group ([11]), it is these very trade-offs and heuristics that make analogy so interesting. They model analogy as a non-deterministic process subject to competing slippage pressures that can shape very different (but valid) solutions. Their work focuses on proportional analogies within closed micro-worlds, dealing e.g. with letter-sequences or table-top place settings. Yet though insightful, it is not immediately clear how a micro-world approach can gain traction on analogies between arbitrary propositions extracted from open-domain texts.

Turney ([22], [23])) works with word-based proportional analogies drawn from SAT-style tests, in which one must find the best analogical match for a given pair of terms (e.g., *jury:verdict*) among a set in which the right answer (e.g., *courier:package*) is mingled with distracters (e.g., *judge:trial*). To avoid a fragile reliance on a knowledge-base of hand-coded representations, Turney employs a distributional model of relational meaning in which a vector space of distributional features is derived from corpus or web text, and smoothed via singular value decomposition (SVD). In his approach, called *Latent Relational Analysis* (LRA), the features are words and phrases that can link the two elements of a relational pair, such as "delivers" for *jury:verdict* and *courier:package*. LRA achieves impressive performance on real SAT analogies, comparable to that of the average human test-taker. Turney shows that the distributional approach is more effective than a pure knowledge-based approach; applying WordNet ([8]) to Turney's dataset of 374 SAT analogies, Veale ([25]) reports a lower precision of 38%-44% using WordNet alone, attaining a bare pass mark. Nonetheless, WordNet-based approaches are much less computationally demanding, hinging on a lightweight measure of semantic similar-

ity that can be efficiently applied on a large scale to tens of thousands of potential analogies.

The *AnalogySpace* model ([19]) sees analogy as a natural outgrowth of a large knowledge-base, in this case the *ConceptNet* project of Liu and Singh ([15], [20]). Comprising facts and generalizations acquired from the template-structured contributions of web volunteers, ConceptNet expresses many relationships that accurately reflect a public, common-sense view on a given topic (from *vampires* to *dentists*), but also many that are idiosyncratic or ill-formed. As in Turney's LRA [22], AnalogySpace builds a representation with reduced-dimensionality in which analogies emerge from the mulching together of perspectives that have deep similarities despite their superficial dissimilarities. AnalogySpace does not concern itself either with complex system analogies or even with proportional analogies between individual propositions, but with the identification of concepts that share analogical similarity (e.g., things that evoke similar feelings and are pleasant or unpleasant in similar ways).

As in AnalogySpace, this current work assumes that analogy occurs within a large knowledge-base of diverse propositions. Our approach is also corpus-based, like LRA, but we do not use representations with reduced-dimensionality, nor is our main goal here the detection of analogies within an existing KB or dataset. Rather, we use analogy as a guide to how the knowledge-base should grow, and provide a wholly symbolic implementation of proportional analogy that efficiently hinges on the simple notion of pragmatic comparability, which we describe next.

3 Learning to Compare, Pragmatically

A taxonomic model of word meaning like WordNet can be used to provide a numeric similarity score for any two terms one cares to compare, no matter how odd the pairing. Budanitsky and Hirst [5] evaluate a menu of WordNet-based similarity functions, and whether one is comparing *prawns* and *protons* or *galaxies* and *footballs*, WordNet can be used to provide a sensible measure of their semantic similarity. Experiments have shown that WordNet-based similarity measures broadly reflect human intuitions ([16], [18], [27]), though such measures are best viewed as relative, to know e.g., that protons are more similar to electrons than to crustaceans.

Yet the biggest advantage to this approach is also its greatest weakness. The space of sensible comparisons is far smaller than the space of possible comparisons, and WordNet can be used to attach a non-zero similarity score to the most ill-judged of comparisons. This may not be a concern if we can trust the client application to only seek similarity scores for terms it has good reason to compare, as when interpretation of the analogy *runner:marathon::oarsman:regatta* requires a comparison of *runner* to *oarsman* and *marathon* to *regatta*. However, speculative matches of *runner:marathon* to hundreds or even thousands of potential analogues in the KB will inevitably result in silly comparisons that no human would ever make, even if they do yield good similarity scores. The situation is exacerbated by lexical ambi-

guity, since the word senses that yield the best scores may not be the intended or most natural senses for the analogy in question.

WordNet-based measures are semantic, objective, and context-free, uninfluenced by subjective and pragmatic considerations. Any measurement typically involves a small number of static category structures, such as *mammal* when comparing *cats* and *dogs*, or *vehicle* when comparing *cars* and *buses*. In contrast, distributed corpus-based approaches implicitly capture the diverse contexts in which we experience two terms/ideas ([28]). For instance, *pirates*, *astronauts* and *cowboys* are semantically similar by virtue of being *human beings*, but are pragmatically similar for a variety of tacit cultural reasons, not least because they represent dashing heroic types that make for "cool" characters in movies and "cool" costumes on Halloween. The distributed approach is successful because we cannot hope to articulate all the reasons why two terms are pragmatically comparable, much less express them as static categories in WordNet.

The coordination "*astronauts* and *cowboys*" suggests that both terms are comparable because they occupy the same context- or task- specific ad-hoc category ([2]). Set-building linguistic constructs provide evidence of subtle pragmatic categories that cannot be lexicalized in WordNet. Parsing such constructs, like lists and coordinations, is the basis for *Google Sets*, a tool that allows Google to perform on-demand set completion ([21]). To obtain our own comparability judgments for generic-level concepts, we harvest all coordinations of bare plurals (e.g., "*cats* and *dogs*" and even "*atoms* and *galaxies*") and of proper names (such as "*Paris* and *London*" or "*Zeus* and *Hera*") from Google's 1T database of web n-grams ([3]). For each pair of coordinated terms, we calculate a similarity score based on the relative depth of their senses and their common hypernym in the WordNet sense hierarchy ([18]), and populate the comparability matrix accordingly. The n-grams link 35,019 unique terms, but each is coordinated with relatively few other terms, so only a tiny fraction of the possible cells in the $35,019 \times 35,019$ comparability matrix will have non-zero similarity scores.

This matrix is symmetric, since we assume that the similarity score for X and Y is the same as the similarity score for Y and X. By finding the cell at the intersection of row X and column Y (or of row Y and column X), a system can quickly look up the pre-compiled similarity score of X to Y. One important reason to encode a similarity function as a pre-compiled matrix is that we wish to re-imagine similarity as a generative phenomenon. Conventional similarity metrics implement a convergent process: given a pair of concepts X and Y, the metric converges on a single, objective score. But such convergent metrics cannot readily be used to generate a divergent set of possible comparisons $\{Y_1 \ldots Y_n\}$ for a given term X ([27]). Likewise, a convergent similarity metric can converge on a single score to represent the similarity underpinning a proportional analogy A:B::X:Y as a function of the similarity of A to X and of B to Y (and perhaps of the latent relationship R_{AB} to R_{XY}). However, a convergent metric cannot be used to generate a set of plausible pairs $\{X_1{:}Y_1 \ldots X_n{:}Y_n\}$ when presented with only half of an analogy, A:B. By using corpus analysis to construct a comparability matrix, and a convergent similarity metric to populate this matrix, a system can generate plausible, divergent comparisons $\{Y_1 \ldots Y_n\}$ for a

given term X simply by reading off the row for X in the matrix. Such a system need not worry about generating dud comparisons, as every non-zero cell in the matrix corresponds to a corpus-attested (if implicit) comparison.

This sparse matrix is compact enough to store in memory, yet contains all of the most plausible comparisons a system is ever likely to consider. The matrix can be used to suggest as well as interpret comparisons. So to suggest sensible analogical substitutions for a term/concept X, we simply read off the row for X in the matrix. Rows can also be intersected to suggest sensible answers for missing elements in proportional analogies, as in:

a) priest : church :: ? : mosque *(A: imam)*
b) church : spire :: mosque : ? *(A: minaret)*
c) chef : recipe :: scientist : ? *(A: formula)*
d) school : bus :: hospital : ? *(A: ambulance)*

In (a), the answer must be comparable to *priest* and coordinated with *mosque*, and is found by looking for the most similar terms to priest among the intersection of the rows for *mosque* and *priest*. So the answer to (a) lies at the intersection of the 3-grams "priests and imams" and "imams and mosques". The matrix yields sensible answers to (b), (c) and (d) in the same way.

This approach is elaborated in section 5, to consider analogies of propositions with a specific relation and a minimum similarity threshold.

4 A Knowledge-Base of Commonplace Generalizations

A bare plural like "dogs" typically denotes the generic use of the category, rather than a specific group of instances ([6]). Thus, the 3-gram "dogs and cats" typically denotes a generic connection between the concepts CAT and DOG at the category level, which suggests that one might be a viable substitute for the other in an analogy. If we replace the coordinator "and" with an appropriate verb, we can determine the generic relationship that links both concepts ([17]). For instance, "dogs *chase* cats" expresses the generalization that, all things being equal and if given the chance, an instance of DOG will chase an instance of CAT. By their very nature, most generalizations are easily falsified, yet they convey defeasible commonsense insights into the workings of the world as seen through stereotypes.

To acquire a large number of generalizations, we can look for generic propositions that predicate over generic uses of concepts. Looking to the Google 3-grams again, we harvest all instances of the template "Xs <verb> Ys". Matches include "birds lay eggs", "scientists conduct experiments" and "chefs create dishes". In all, we find 158,911 matches for "Xs <verb> Ys" in the Google 3-grams. But so simple a template will inevitably retrieve a great deal of noise: some of the retrieved matches will be syntactically ill-formed (e.g., the verb is part of a phrasal verb, or has a complex sub-categorization frame that does not fit into a 3-gram, as in "priests tell parishioners [that]") while many more will express generalizations that are simply false (e.g., "mammals lay eggs"), obscene (this is web content, after all), or not

worth adding to a knowledge-base. One cannot be totally objective when dealing with generalizations, so there is no official count for the number of valid generalizations to be mined from the Google n-grams (or any other open-domain source). What is added to a KB is determined by domain relevance, tolerance for imprecision and personal taste. Individual consideration of all 158,911 matches suggests that only one in eight of these raw matches (or 21,258 of 158,911 to be precise) yields a generalization that is sound enough to merit a place in a knowledge-base. A full analysis of this KB of n-gram-derived generalizations is presented in section 6.

We can use a more sophisticated extraction process, or target longer n-grams for more complex generalizations. But for our current purposes, these matches, noise and all, make an ideal test-set. We view our task as the dynamic ranking of these 158,911 matches so that the 21,258 valid generalizations that are competing with the remaining 137,653 rejects for entry to the KB (i.e., 158,911 - 21,258) are prioritized and brought to the front of the knowledge-acquisition queue. Success with this noisy dataset will indicate the potential for analogy to expedite the results of more sophisticated mining algorithms with a lower tolerance for noise.

5 Analogy-Guided Knowledge Acquisition

The knowledge-acquisition queue contains all those propositions of yet-unproven value that are extracted from corpora or volunteered by web users. In an ongoing KB effort, this queue may be constantly growing, and like celebrities at a nightclub, new arrivals may jump the queue if perceived as a better fit for the KB. Of course, when the knowledge-base is initially empty there is no analogical basis for preferring one queued proposition over another, as there are no propositions in the KB with which to form an analogy.

Creating a new knowledge-base requires a knowledge engineer to walk through the queue, accepting propositions that are valid and rejecting those that are not. This task is made increasingly more efficient by analogy: as new propositions are added, the system looks for analogies between any valid proposition in the KB and any unseen proposition remaining on the queue. As the KB grows, so too will the number and diversity of these potential analogies. The system sorts the queue dynamically, after each new addition to the KB, ranking candidates by the number of potential analogies that can link them to the growing KB. In a growing queue, ill-fitting propositions will always be pushed to the back.

We use simple (and flat) structure-mapping to perform analogical matching. For propositions of the form $pred(X,Y)$, structure-mapping is simple to apply, both to flat and to recursively-nested structures:

> $pred^S(X,Y)$ is a match for $pred^T(A,B)$ *iff* $pred^S = pred^T$ so that X is mapped to A and Y is mapped to B.

At the heart of any structure-mapping problem then is a series of one of more proportional analogies of the form A:B::X:Y, where the mapping of A to B and of X

to Y is suggested by, and in a sense guaranteed by, the identically of the relation that connects A to B and the relation that connects X to Y. Predicate identicality is a standard meaning-preserving constraint in structure mapping (see [9], [7]), one that also reduces the search space of the mapping problem. Structure-mapping is typically applied to nested structures whose isomorphism strongly suggests the comparability of literal predicate arguments. For shallower structures, as with our simple generalizations, we can also ensure the comparability of arguments by demanding corpus evidence of their substitutability. Generalizations such as $create(artist, artwork)$ and $create(chef, dish)$ can be mapped in an analogy because both use the predicate $create$ with the same arity, while the 3-grams "*artists and chefs* and "*artworks and dishes* also suggest that *artists* are comparable to *chefs* while *dishes* are comparable to *artworks*.

Comparability ensures sensibility when interpreting analogies and efficiency when generating analogies. So to find a match for the proposition $pred(A, B)$, we consider only those KB propositions $pred(X, Y)$ that have the same predicate *pred* such that A is comparable to X and B is comparable to Y. Since a system can read off the set of comparable terms for A and for B from the comparability matrix, it can divergently generate all sensible analogues of $pred(A, B)$ within a given similarity threshold and look for them in the KB. For instance, at a similarity threshold of 80%, we generate the analogue $pred(X, Y)$ only if A and X have a similarity of 80% or more in the comparability matrix, and if the similarity of B and Y is also at or above this threshold. At a threshold of 0%, we allow all analogies between propositions with the same predicate and arity.

Analogies are therefore proposed using a *generate-and-test* rather than a *find-and-match* approach, so the system can suggest meaningful analogies that the KB does not yet (but perhaps should) support. Clearly, fewer analogies are generated at higher similarity thresholds, while many more are generated as we lower the threshold. There is an inevitable recall-versus-precision trade-off here: as the similarity threshold is lowered, we open the door to more creative analogies with greater semantic distance, but we may also reduce the average quality of the larger pool of analogies that is proposed. In the next section we consider the effect of the similarity threshold on analogue retrieval, and its knock-on effect on the workings of analogy-guided knowledge-acquisition.

6 Empirical Evaluation

This chapter has put forth two broad claims that require empirical validation. The first concerns our corpus-based model of comparability, which assumes – for the sake of naturalness and efficiency – that the space of sensible comparisons is sparse. We verify that this sparseness does not impair coverage, to show that comparability provides a robust and compact basis for aligning like with like. The second concerns the efficacy of analogy-guided acquisition, which assumes that candidate additions to the KB should be sorted according to analogical fit. We show that a KB grows

fastest when this is so, and slowest in the worst-case scenario when this fitness metric is perversely ignored.

6.1 The Reliability of Comparability

An analysis of coordination patterns in the Google 3-grams fills the comparability matrix with similarity scores for 35,019 unique terms. Any pair of terms is likely to yield a non-zero score using a WordNet-based measure, yet n-gram analysis finds just 1,363,184 pairs, or just 1.36×10^6 pairs out of a possible $1,225 \times 10^6$ pairings. With a density of approx. 0.1% ($1.36/1,225$), the matrix is sparse enough to hold in memory, but one can ask whether it is too sparse to be a general model of comparability. So to see if it provides sufficient coverage for robust category-level reasoning, we use it to replicate the category-formation experiments of [1].

The authors of [1] select 214 words from 13 different categories in WordNet. Using query patterns in the style of Hearst ([10]) to retrieve informative text fragments for each word from the web, they then harvest a large body of features for each word. These features include attributes, such as TEMPERATURE for coffee, and attribute values, such as FAST for car. In all, they harvest a set of approx. 60,000 web features for the 214-word dataset, and use CLUTO ([13]) to automatically group the words into 13 clusters on the basis of their web-harvested features. The *purity* of a cluster is a measure of its homogeneity, and of the extent to which the members of the cluster all belong to the same category. When the average purity of a set of clusters is 1.0, this indicates that the clusters faithfully replicate the category boundaries inherent in the original data (which, in this case, are the boundaries of WordNet's categories). These 13 CLUTO-built clusters have a purity of 0.85 relative to WordNet's own categories, which represents an 85% replication of WordNet's structure.

We replicate the experiment using the same 214 words and 13 WordNet categories. However, rather than harvesting web features for each word, we use the corresponding rows of our comparability matrix. Thus, the features for "chair" are the comparable terms for "chair", that is, the set of X such that either "chairs and Xs" or "Xs and chairs" is a Google 3-gram. We ignore the similarity scores in the matrix, as these were produced using WordNet, but we do treat every word as a feature of itself, so e.g., CHAIR is a feature of "chair". The sparse matrix yields a much smaller set of features – 8,300 in total for all 214 words. CLUTO builds 13 clusters from these features, and achieves a cluster purity of 0.934 relative to WordNet. The space of sensible comparisons is indeed a compact and precise means of representing the semantic potential of words.

6.2 Analogical Fit *as a Guide to Knowledge Acquisition*

For evaluation purposes we annotate each of the 158,911 generic propositions from the 3-gram dataset as *valid* (if it is sensible to add it to a KB) or *invalid* (if it should be rejected from any KB). No attempt at global coherence is made: a proposition is hand-tagged as valid if it has the ring of commonsense truth, even if only as a stereotype. Different engineers may quibble about the wisdom of individual propositions, and might build slightly different KBs of their own. We consider the effect of different choices at the end of this section.

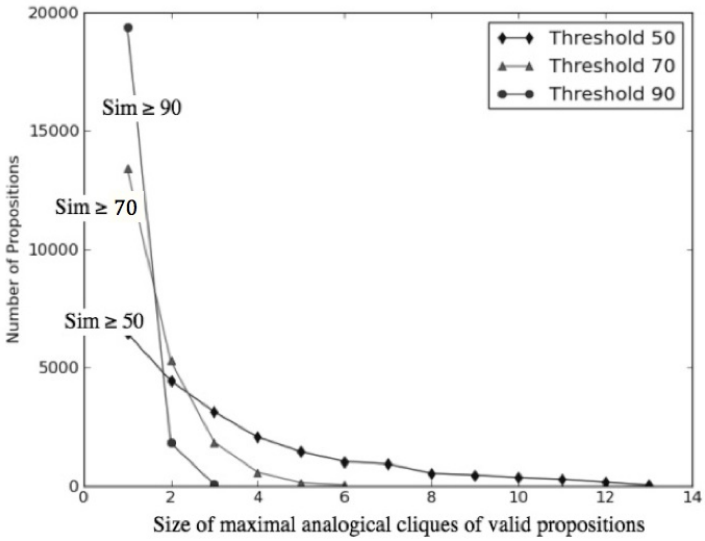

Fig. 1 Distribution of maximal analogical cliques of propositions tagged as valid, using three similarity thresholds (50%, 70%, 90%) for analogical matching.

In all, we tag 21,258 propositions as valid, or about 1 in 8. Analogy makes neighbors of similar propositions, and a clique analysis (see [4]) reveals the neighborhood structure imposed by analogy on this dataset. An *analogical clique* is a set of propositions that are all mutually connected by analogy at a given similarity threshold. An analogical clique is maximal if no valid proposition can be added to make it larger. Figure 1 shows the range of valid propositions in maximal analogical cliques of varying sizes. As the similarity threshold is lowered, the number of larger analogical cliques increases significantly.

We use this hand-tagged set of valid propositions as the basis of an oracle for simulating different knowledge-acquisition scenarios. We start with a KB of 1,000 *valid* propositions, randomly chosen from those annotated as valid. Figure 2 shows the growth of this seed KB as the queue of the remaining 157,911 candidates is processed.

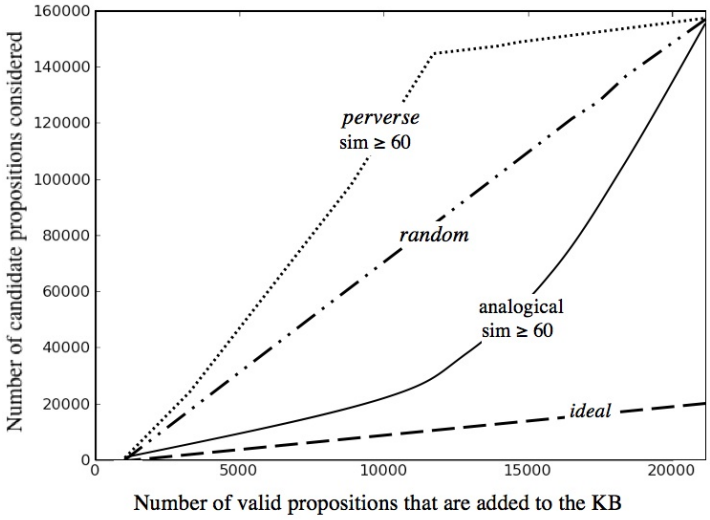

Fig. 2 A graph of the number of queued candidates that must be considered (y-axis) to grow the KB to a given number of valid propositions (x-axis). A similarity threshold of 60% is used for making analogies.

The *ideal* case shows the growth of the KB when every candidate served from the queue is valid: so we need process just 19,000 additional candidates with a hit-rate of 100% to grow the KB to a size of 20,000. The *random* case shows the baseline growth of the KB when the queue is unsorted. The *analogical* case shows growth when the queue is sorted in descending order of the number of possible analogies from each proposition on the queue to valid propositions in the KB. The *perverse* case aims to model the worst-case scenario, by sorting the queue in ascending order, so candidates with the least analogical potential are processed first. Note that any non-ideal curve will be anchored at (1000,0) and (21,258, 158,911) since, lacking perfect information, a system must consider all 158,911 candidate propositions to identify every valid proposition. Each curve varies in the middle part of its trajectory between these two anchor points, turning away from the ideal line as valid propositions are encountered with diminishing frequency among the remaining candidates.

A similarity threshold of 60% on the depth-based WordNet metric of Seco, Veale and Hayes ([18]) was used for these experiments, though any WordNet-based similarity metric will suffice ([5]). The *perverse* case is clearly worse than the random baseline, while the *analogical* case is closer to ideal growth. Figure 3 shows that as the similarity threshold is lowered, and the potential for analogy is increased, we see the performance of the analogy-sorted queue move even closer to the *ideal* case. However, even at a 0% threshold, Figure 3 shows there is a considerable gap between the system‘s performance and the ideal case. A 0% threshold turns the selection of valid propositions into a *predicate-popularity contest*: the propositions with predicates (and matching arities) that are most representative of the valid propo-

sitions already in the knowledge-base will be favored over those with previously unseen or rare predicates. This is a reasonable but imperfect strategy that shows the limits of analogical ordering, and no matter how low one sets the similarity threshold, a system cannot close the gap with the ideal case.

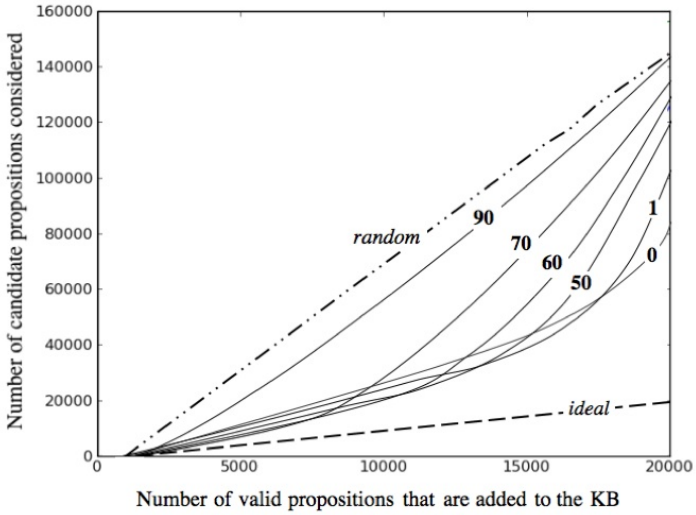

Fig. 3 Performance of analogy-guided acquisition at different similarity thresholds for making analogies.

Each valid proposition in the knowledge-base can viewed as marking an area of semantic space in which semantically similar, and equally valid, propositions might be found. The size of this space is dictated by the similarity threshold we choose: a high-threshold shrinks the space around each landmark proposition, reducing the number of analogical cliques that might be built around that proposition, and resulting in higher-precision but reduced recall for the analogical inferences we may draw. Conversely, a low threshold expands the space around each valid proposition, increasing the recall but diminishing the precision of our analogical inferences regarding the validity of novel propositions. Each threshold thus represents a compromise between recall and precision; the point on each curve where the curve takes a markedly upward turn is the point where this compromise tilts from effective to ineffective. So few analogies are permitted by a 90% similarity threshold that performance of the analogy-sorted queue is close to that of the random baseline. However, each successive lowering of the threshold, all the way down to 1%, shows obvious gains. At a threshold of 0% – which permits pure structure-mapping with no comparability constraints– this weak compromise produces mixed results. Remember, at a 0% threshold our analogies are no longer constrained to employ corpus-supported substitutions, and analogical matching becomes simple predicate and arity match-

ing. At a 0% threshold, the system favors those propositions whose combination of predicate and arity are observed most frequently in the existing knowledge-base.

As shown in Figure 4, lowering the threshold for analogy makes the *perverse* case even worse, especially at the 0% similarity threshold. Recall that in the perverse case –in which we attempt to model the worst-case scenario – the propositions that are deemed to have no analogical potential are considered first. However, the higher the similarity threshold, the greater the number of propositions that the system will deem to have no analogical potential. Many propositions that have some analogical potential (at a lower similarity threshold) will be incorrectly deemed to have none, and processed first anyway. As the similarity threshold is lowered, the perverse case ensures that propositions with even a hint of analogical potential are processed last. Each graph for the *perverse* case thus shows two stages: an early stage where candidate propositions that are deemed to have no analogical potential are considered first, and a second stage where candidates with some potential are considered in reverse order of this potential. As Figure 4 shows, each lowering of the similarity threshold for analogy moves candidates from the first to the second stage.

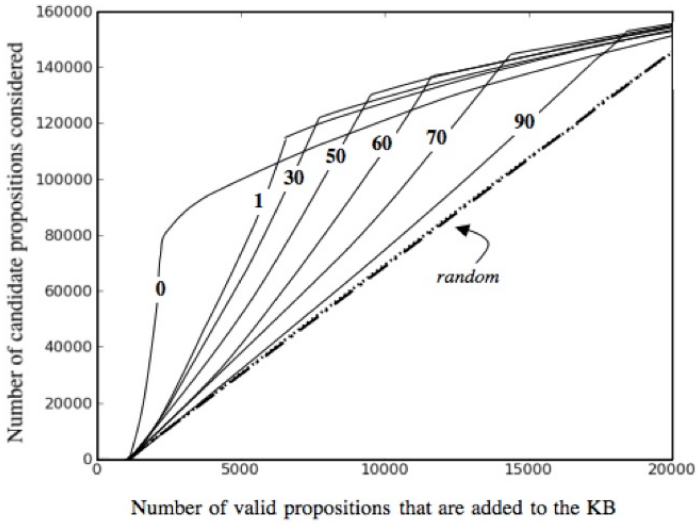

Fig. 4 Worst-case acquisition scenarios (in the *perverse* case) at varying similarity thresholds for making analogies. The queue is sorted so that propositions with the most analogical potential are served last.

But the real test of analogy-guided acquisition is how much time it saves a knowledge-engineer. Imagine we want to double the current size of our KB by adding only valid propositions from the queue of candidate additions. Figure 5 graphs the speed-up factor achieved when doubling the KB size, relative to the random baseline of an unsorted queue, at different similarity thresholds.

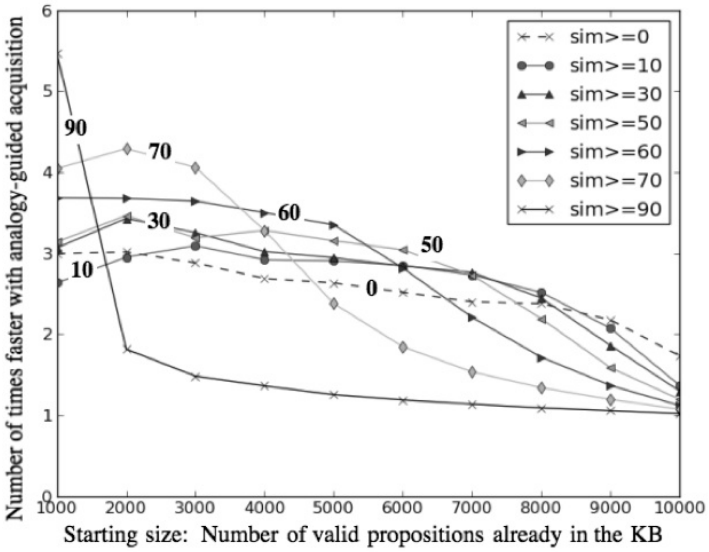

Fig. 5 Efficiency savings when using analogy-guided acquisition. A system that uses analogy-guided acquisition with a given similarity threshold will double the initial size (X valid propositions) of its knowledge-base Y times faster than a system that does not use analogy-guided acquisition (the random case). Thus, e.g., a system that uses a similarity threshold of 10% will double its knowledge-base from X=2000 valid propositions to 4000 valid propositions Y=3 times faster than a system that does not use analogy-guided acquisition. These efficiency savings dwindle as the initial size (X) of the knowledge-base approaches 50% of its final size, as a system must then consider *all* incoming propositions if it is to double its size.

Figure 5 shows that using analogy as a guide, doubling the KB size from 3000 to 6000 valid propositions is 4 times faster than using an unsorted queue, at a similarity threshold of 70%. Doubling from 5000 to 10000 is 3.5 times faster at a similarity threshold of 60%. This speed-up declines as the threshold is raised, and at 90% similarity there is practically no speed-up at all, because so few analogies are made at this level. But the speed-up also declines as the KB grows in size, due to the *knowledge dilution effect*.

As valid propositions are removed from the pool of candidates and added to the KB, the concentration of valid propositions in the remaining pool is diluted. Initially around 1 in 8, the *dilution rate* of valid to invalid candidates can drop to 1 in 40 for the last 1000 valid propositions. Analogy-guided retrieval gives priority to those candidates with the most analogical potential, pushing those with little or no potential to the back of the queue. As the KB approaches its maximum size and valid propositions are crowded out by higher levels of noise, they are less differentiated by their analogical potential, causing overall performance to regress toward the random baseline.

Figure 5 suggests that a mixed acquisition strategy makes the most sense: analogy-guided acquisition should use higher similarity thresholds in the earlier

stages of KB construction, both to achieve higher throughput and so that system suggestions can be explained to the user in the form of obvious, semantically-grounded analogies. As the KB grows, acquisition can shift to incrementally lower similarity thresholds to offset the effects of knowledge-dilution. As low-hanging fruit is harvested, a system must rely on increasingly creative (and perhaps unsafe) analogies to maintain an efficient acquisition process.

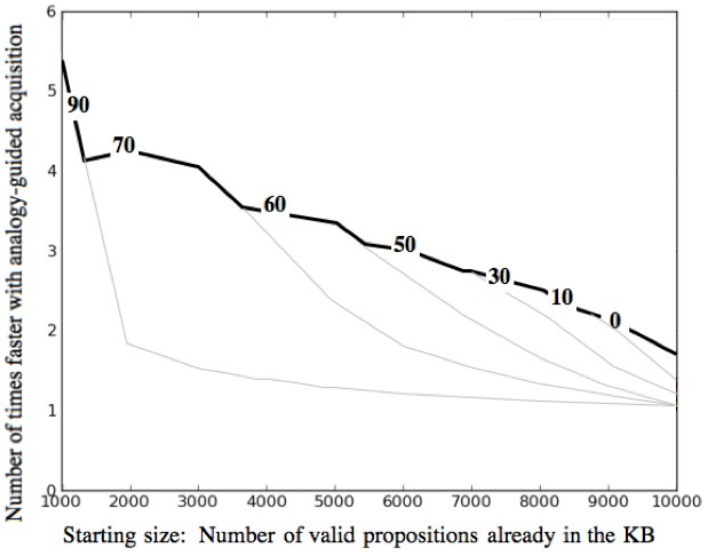

Fig. 6 Best performance is achieved by gear-changes from higher to lower similarity thresholds.

Higher-levels of similarity produce safer and more obvious analogies, but yield unsustainable efficiency gains. Figure 6 shows the gear changes that are needed to maximize the throughput of the acquisition process.

But can subjective differences in opinion about which propositions are valid or invalid alter the dynamics of analogy-guided acquisition? To find out, we imagine an extreme case: rather than use our manual annotations of propositions as valid or invalid, we randomly label 21,258 propositions from our pool of 158,911 n-gram candidates as *valid*, and label all others as *invalid*. This yields a randomized candidate pool with the same 1-in-8 distribution of *valid* propositions.

Figure 7 shows the resulting collapse in performance at all similarity thresholds on the fully randomized KB. We see virtually no analogical structure at all in this KB, and the *perverse*, *random*, and *analogical* cases become indistinguishable. This illustrates that a knowledge-base is much more than a jumble of random facts and generalizations: the more coherent our knowledge, the greater the potential for analogy and the bigger the role of analogy-guided acquisition.

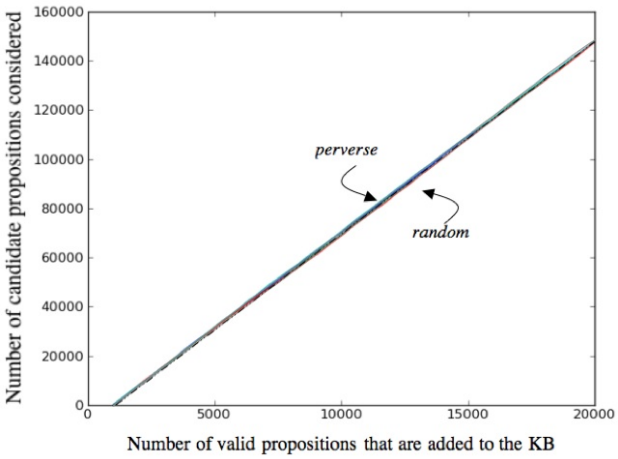

Fig. 7 Change in performance of analogy-guided acquisition on a randomly-constructed KB.

7 Conclusions and Future Work

Large knowledge-bases are subject to *neighborhood effects* ([29]) since propositions add more value to a KB when they interact effectively with others. This chapter has argued that analogy and comparability are an effective means of predicting precisely this kind of neighborly interaction.

More specifically, we have presented a means of searching the pool of potential additions to a knowledge-base using analogy-guided best-first search. Queued additions that suggest the most analogies to valid propositions in the KB (at a specific similarity threshold) are moved to the front of the queue. A knowledge-base that strives to model an evolving world will always be in need of new propositions, and analogy-guided acquisition is assumed to occur within a knowledge-base that is constantly growing. That is, analogy-guided acquisition works best when used with a queue that is constantly receiving new candidate propositions from external sources. As new candidates arrive, the most promising are ushered to the front of the queue based on their likelihood of forming productive analogies and analogical cliques with the propositions that already reside in the KB.

We define analogy simply, in terms of structured comparability, which is a corpus-trainable pragmatic version of semantic similarity. Comparability is simultaneously a more generative and a more restrictive notion than semantic similarity, and it is this combination of generativity and restrictiveness that makes sensible analogical mapping efficient on a large scale. A generative measure of semantic similarity can be used to both evaluate and suggest analogies. We have shown that analogy is a practical predictor of a candidate's marginal value to a KB, even at very low thresholds of similarity.

In future work we must also examine the performance of analogy-guided acquisition on more complex types of propositional content than that considered here. In this vein, a productive starting point is the large set of implicit clique structures that naturally coalesce within the coordination graph that underpins our notion of corpus-driven pragmatic comparability. Recall that this graph contains an edge between two terms X and Y if corpus evidence (i.e., the Google 3-grams) suggests that X and Y are comparable. More specifically, an edge links X to Y if either of the 3-grams "Xs and Ys" or "Ys and Xs" can be found within the Google web n-grams database. These edges form complete sub-graphs of k nodes, or *k-cliques*, in which each node is connected by a corpus-attested edge to the $k-1$ other nodes of the *k*-clique ([4]). A clique is maximal if it is not a proper-subset of another clique in the same graph. Each maximal *k*-clique within the coordination graph represents a tight cluster of highly-interrelated knowledge, of a kind that should be treated as a single whole, as a pragmatic category of sorts. The distribution of maximal cliques in the coordination graph for different sizes of k (clique size) is shown in Figure 8.

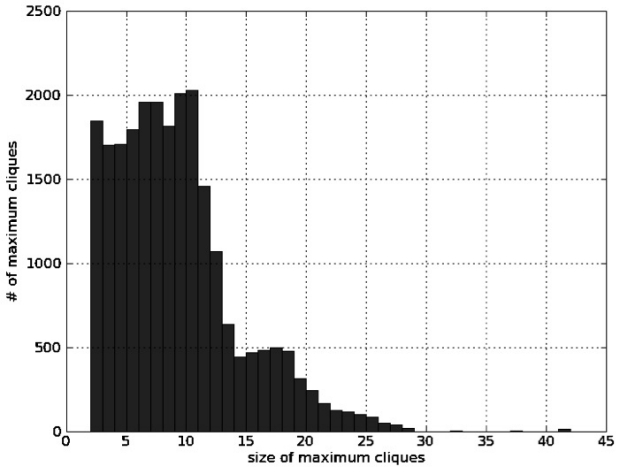

Fig. 8 Cliques of different sizes in the coordinations graph mined from the Google 3-grams.

Consider a simple example for $k = 3$. The 3-gram-derived coordination graph contains the following pair of 3-cliques (and many more besides, as shown in Figure 8): {*scientist, laboratory, experiment*} and {*artist, studio, exhibition*}. But the coordination graph also contains edges that link *scientist* to *artist*, *laboratory* to *studio*, and *exhibition* to *experiment*. As such, if a KB contains propositions to label each edge in the 3-clique {*scientist, laboratory, experiment*} with an apt predicate, it can project these labels/predicates onto the 3-clique {*artist, studio, exhibition*} and thereby form a truly systematic analogy (of the kind discussed in [9], [7], [24]). This situation is illustrated in Figure 9.

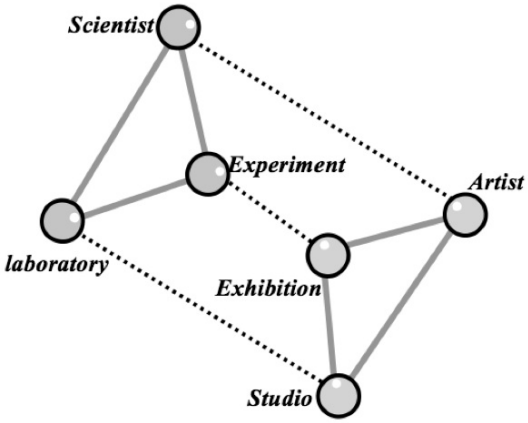

Fig. 9 A potential analogy between a pair of 3-cliques in the coordination graph.

In other words, a KB system can do more that seek to prioritize the addition of propositions that are analogous with previously acquired propositions. A system can actively seek to acquire propositions that allow it to build ever more systematic mappings between cliques of concepts, and between cliques of propositions ([26]).

We shall additionally look to exploit the generative capabilities of cliques and of the *generate-and-test* approach to analogy-making. This should allow the analogy-guided acquisition process to propose its *own* additions to the KB, over and above those in the queue of candidates extracted from corpora. For analogy has an intriguing role to play in shaping as well as recognizing valid knowledge.

Acknowledgements This research was supported by the WCU (World Class University) program under the National Research Foundation of Korea, and funded by the Ministry of Education, Science and Technology of Korea (Project No: R31-30007).

References

1. Almuhareb A. and Massimo P.: Attribute-Based and Value-Based Clustering: An Evaluation. Proceedings of EMNLP, Empirical Methods in NLP, 158 165. (2004)
2. Barsalou, L.W.: Ad hoc categories. Memory and Cognition, 11:211–227. (1983)
3. Brants, T. and Franz, A.: Web 1T 5-gram Version 1. Linguistic Data Consortium. (2006)
4. Bron, C. and Kerbosch, J.: Algorithm 457: Finding all cliques of an undirected graph. Communcations of the ACM 16(9). (1973)
5. Budanitsky, A. and Hirst, G.: Evaluating WordNet-based Measures of Lexical Semantic Relatedness. Computational Linguistics, 32(1):13–47. (2006)
6. Carlson, G. N. and Pelletier, F. (eds.): The Generic Book. University of Chicago Press. (1995)
7. Falkenhainer, B., Forbus, K. D., and Gentner, D.: Structure-Mapping Engine: Algorithm and Examples. Artificial Intelligence, 41:1–63. (1989)

8. Fellbaum, C.: (ed.). WordNet: An electronic lexical database. Cambridge, MA: MIT Press. (1998)
9. Gentner, D.: Structure-mapping: A Theoretical Framework. Cognitive Science 7:155–170. (1983)
10. Hearst, M.: Automatic acquisition of hyponyms from large text corpora. Proceedings of the 14th International Conference on Computational Linguistics, 539–545. (1992)
11. Hofstadter, D.R.: Tracking sentiment in mail: how genders differ on emotional axes. Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought. New York, NY: Basic Books. (1995)
12. Holyoak, K. J. and Thagard, P.: Mental Leaps: Analogy in Creative Thought. Cambridge, MA: Basic Books. (1995)
13. Karypis, G.: CLUTO: A clustering toolkit. Technical Report 02-017. University of Minnesota. http://www-users.cs.umn.edu/ karypis/cluto/ (2002)
14. Lenat, D. and Guha, R.V.: Building Large Knowledge-based Systems. New York, NY:Addison Wesley. (1990)
15. Liu, H. and Singh, P.: ConceptNet: A Practical Commonsense Reasoning Toolkit. BT Technology Journal, 22(4):211–226. (2004)
16. Miller, G. A. and Charles, W.G.: Contextual correlates of semantic similarity. Language and Cognitive Processes 6(1):1–28. (1991)
17. Nakov, P. and Hearst, M.: Using verbs to characterize noun-noun relations. Artificial Intelligence: Methodology, Systems, and Applications, 233–244. (2006)
18. Seco, N., Veale, T. and Hayes, J.: An intrinsic information content metric for semantic similarity in WordNet. Proceedings of ECAI-2004, the 16th Annual Meeting of the European Association for Artificial Intelligence. (2004)
19. Speer, R., Havasi, C. and Lieberman, H.: AnalogySpace: Reducing the Dimensionality of Common Sense Knowledge. Proceedings of the 23rd AAAI Conference on Artificial Intelligence. (2008)
20. Singh, P.: The public acquisition of commonsense knowledge. Proceedings of AAAI Spring Symposium on Acquiring (and Using) Linguistic (and World) Knowledge for Information Access. Palo Alto, CA. (2002)
21. Tong, S. and Dean, J.: System and methods for automatically creating lists. US Patent 7,350,187 (granted to Google, March 25, 2008).
22. Turney, P. D.: Measuring semantic similarity by latent relational analysis. Proceedings of IJCAI-2005, the 19th International Joint Conference on Artificial Intelligence, Edinburgh, 1136–1141. (2005).
23. Turney, P.D.:A uniform approach to analogies, synonyms, antonyms, and associations, Proceedings of the 22nd International Conference on Computational Linguistics (COLING 2008), Manchester, UK, 905-912. (2008).
24. Veale, T. and Keane, M.T.: The Competence of Sub-Optimal Structure Mapping on 'Hard Analogies'. Proceedings of ICJAI-1997, the 15th International Joint Conference on Artificial Intelligence, Nagoya, Japan. (1997)
25. Veale, T.: WordNet sits the SAT: A knowledge-based approach to lexical analogy. Proceedings of ECAI-2004, the 16th Annual Meeting of the European Association for Artificial Intelligence. (2004)
26. Veale, T. and Li, G.:Ontological cliques-analogy as an organizing principle in ontology construction. Proceedings of The International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, Madeira. (2009).
27. Veale, T. and Li, G.:Creating Similarity: Lateral Thinking for Vertical Similarity Judgments. In Proceedings of ACL 2013, the 51st Annual Meeting of the Association for Computational Linguistics, Sofia, Bulgaria (2009).
28. Weeds, J. and Weir, D.: Co-occurrence retrieval: A flexible framework for lexical distributional similarity. Computational Linguistics, 31(4):433–475. (2005)
29. Weigher, J.C. and Zerbst, R. H.: The Externalities of Neighborhood Parks: An Empirical Investigation. Land Economics 49(1):99–105. (1973)