Cross-cutting Models of Distributional Lexical Semantics

Joseph S. Reisinger*
The University of Texas at Austin
joeraii@cs.utexas.edu

Doctoral Dissertation Proposal

Supervising Professor: Raymond J. Mooney

June 17, 2010

Abstract

In order to respond to increasing demand for natural language interfaces—and provide meaningful insight into user query intent—fast, scalable lexical semantic models with flexible representations are needed. Human concept organization is a rich epiphenomenon that has yet to be accounted for by a single coherent psychological framework: Concept generalization is captured by a mixture of prototype and exemplar models, and local taxonomic information is available through multiple overlapping organizational systems. Previous work in computational linguistics on extracting lexical semantic information from the Web does not provide adequate representational flexibility and hence fails to capture the full extent of human conceptual knowledge. In this proposal I will outline a family of probabilistic models capable of accounting for the rich organizational structure found in human language that can predict contextual variation, selectional preference and feature-saliency norms to a much higher degree of accuracy than previous approaches. These models account for cross-cutting structure of concept organization—i.e. the notion that humans make use of different categorization systems for different kinds of generalization tasks—and can be applied to Web-scale corpora. Using these models, natural language systems will be able to infer a more comprehensive semantic relations, in turn improving question answering, text classification, machine translation, and information retrieval.

*This work was supported by an NSF Graduate Research Fellowship and a Google Research Award. Experiments were run on the Mastodon Cluster, provided by NSF Grant EIA-0303609.

CHAPTER 1

Background and Related Work

In this proposal I argue for a strong synthesis of psychological models of concept organization and computational linguistic models of distributional lexical semantics, focusing particularly on modeling context-dependent variation in word meaning. Concepts and meanings in human language are organized in terms of complex assemblies of properties or features and exhibit significantly richer structure than can be accounted for with traditional lexical semantics models [93].

Humans categorize objects using multiple orthogonal taxonomic structures, where generalization depends critically on what features are relevant to the particular structure. For example, foods can be organized in terms of their nutritional value (high in fiber) or situationally (commonly eaten for Thanksgiving). Furthermore there is significant evidence for overlapping categorization systems in Wikipedia and WordNet (e.g. people are organized by occupation or by nationality). The effects of these overlapping categorization systems manifest themselves at the lexical semantic level [63], implying that lexicographical word senses and traditional computational models of word-sense based on clustering or exemplar activation are too impoverished to capture the rich dynamics of word usage.

To model these phenomena, I introduce a unified set of probabilistic methods based on *cross-cutting categorization* [100], using them to generalize traditional *vector-space* or *distributional* models of lexical semantics [18, 26, 68, 98, 109]. Cross-cutting categorization finds multiple feature subsets (categorization systems) that produce high quality clusterings of the data. For example words might be clustered based on their part of speech, or based on their thematic usage. Context-dependent variation in word usage can be accounted for by leveraging multiple latent categorization systems. In particular, cross-cutting models can be used to capture the *microstructure* of word relatedness, breaking up word features into multiple categorization systems and then computing similarity separately for each system.

Computing semantic relatedness is an important foundational task in distributional lexical semantics, underlying models of word similarity [18], selectional preference [91] and lexical substitution [59]. However, the relatedness is not a globally consistent metric as it violates symmetry (e.g. people have the intution that *China* is more similar to *North Korea* than *North Korea* is to *China*) as well as the triangle inequality (e.g. the sum of distances from *bat* to *club* and *club* to *association* is less than the distance from *bat* to *association*) [32, 110]. Violations of the triangle inequality can be resolved by first breaking up words into senses [88], or using exemplar models of meaning [22], but asymmetry can only be resolved through the use of multiple conditional similarity metrics.

Moving beyond consistency violations, word relatedness for a given pair implicitly defines a typed relation between that pair that may not at all be similar to the relations between similar words. For example *wine* and *bottle* are similar and *wine* and *vinegar* are similar, but it would not be reasonable to expect that the features governing such similarity computations to overlap much, despite all three words occurring in similar documents. The aim of this proposal is to study the application of cross-categorization to find coherent feature subsets that implicitly define meaningful relations, resulting in vector-valued word relatedness.

The particular cross-categorization models proposed here build on my previous work on (1) multi-prototype models of lexical semantics [88] and (2) model-based feature weighting [89]. The proposed work focuses on two areas: (1) fitting cross-cutting categorization models [57, 100] with overlapping clustering structures word occurrence data, determining which subsets of a word's features account for significant variation in meaning, and (2) identifying and removing irrelevant features and occurrences on a per-word basis, yielding robust models. Capturing the multiple overlapping clustering structure of natural concepts leads to improvements on a range of Natural Language Processing (NLP) tasks: question answering [108], unsupervised semantic parsing [82], query intent classification and expansion [41], coreference resolution [34] and textual entailment [106].

The remainder of this chapter summarizes relevant background work in Pyschology (§1.1) and Linguistics (§1.2).

1.1 Psychological Models of Concepts

Psychological models of concepts can be roughly divided into two classes [63]:

1. *Prototype* models represented concepts by an abstract prototypical instance, similar to a cluster centroid in parametric density estimation [3].

2. *Exemplar* models represent concepts by a concrete set of observed instances, similar to nonparametric approaches to density estimation in statistics [4].

Tversky and Gati [110] famously showed that conceptual similarity violates the triangle inequality in the case of polysemous words; e.g. the pair *bat*→*association* is farther apart than the sum of *bat*→*club* and *club*→*association*, lending evidence for exemplar models in psychology [33, 53, 94]. Exemplar models have been previously used for lexical semantics problems such as selectional preference [22] and thematic fit [112]. Individual exemplars can be quite noisy and hence it is common to average over a set of *activated* exemplars in order to improve robustness. However, activating too many exemplars also increases noise by introducing irrelevant features [23].

Voorspoels et al. [115] demonstrate the surperior performance of exemplar models for *concept combination* (e.g. "metal spoon"), suggesting their use in computational lexical semantics when contextual information is available. In general exemplar models are better suited to address polysemy and contextual variation than prototype models. Indeed, experimental evidence suggests that although polysemous words share the same lexical representation, their underlying senses are represented separately, as priming a word in one sense interferes with using it in another, even when the senses are related [44, 45].

Models of conceptual organization are ultimately grounded in some feature space as with vector-space lexical semantic models. Semantic feature production norms—i.e., what features people most often report as salient to a given stimulus concept— have been studied extensively in psychology as one way to understand human concept organization and categorization [60]. However, asking people to report on salient features of specific categories exhibits obvious bias towards discriminative properties that are easy to articulate (e.g. “lions are ferocious” rather than “lions are typically more heavily built than leopards”). Nevertheless without direct access to actual features used in mental representations of concepts, human production norms are the best way to evaluate automatic property generation in terms of relevance.

Moving beyond simple prototype or exemplar models, humans use overlapping taxonomies to organize conceptual information in many domains [93]; i.e. foods can be organized situationally, *breakfast food*, *dinner food*, *snack*, etc, or by their type, *dairy*, *meat*, etc. Each organization system may have different salient features and hence yield different patterns of similarity generalization [cf. 37]. For example, Shafto and Coley [99] find that when reasoning about the anatomical properties of animals relies on taxonomic categories such as *mammals* or *reptiles* whereas reasoning about disease transmission relies on ecological categories such as *predator* and *prey*. One of the main contribution of this proposal will be to explore the degree to which the overlapping categorization structure of concepts can account for generalization and variation in word meaning and help overcome feature noise.

1.2 Distributional Lexical Semantics

Word meaning can be represented as high-dimensional vectors inhabiting a common space whose dimensions capture semantic or syntactic properties of interest [e.g. 68, 98, 109]. Such *vector-space* representations of meaning induce measures of word similarity that can be tuned to correlate well with measurements made by humans. Previous work has focused on designing feature representations and semantic spaces that capture salient properties of word meaning [2, 26, 19], directly leveraging the *distributional hypothesis*, i.e. that similar words appear in similar contexts [35, 62, 50, 76]. Vector spaces are commonly derived from (1) word collocations [98], (2) syntactic relations [68], (3) structured corpora (e.g. Gabrilovich and Markovitch [26]) or (4) latent semantic spaces [25]. Automatically judging the degree of semantic similarity between words is an important task useful in text classification [6], information retrieval [96], textual entailment, and other language processing tasks.

In addition to leveraging the distributional hypothesis, previous work on lexical semantic relatedness has focused on mining monolingual or bilingual dictionaries or other pre-existing resources to construct networks of related words [1, 84]. This approach tends to have greater precision, but depends on hand-crafted dictionaries and cannot, in general, model sense frequency [12]. The vector-space approach is fundamentally more scalable as it does not rely on specific resources and can model corpus-specific sense distributions. However, it can suffer from poor precision, as thematically similar words (e.g., *singer* and *actor*) and antonyms often occur in similar contexts [51]. Thus, vector-space models are typically posed as identifying thematically *related* words, rather than synonyms [2].

Recently Reisinger and Mooney [88] used unsupervised word-sense discovery coupled with traditional measures of semantic similarity to derive a *multi-prototype* model of word meaning. Word-sense discovery has been studied by number of researchers [1, 98]. Most work has also focused on corpus-based distributional approaches, varying the vector-space representation, e.g. by incorporating syntactic and co-occurrence information from the words surrounding the target term [77, 71]. By employing multiple prototypes per word, vector space models can account for homonymy, polysemy and thematic variation in word usage, like exemplar models, while limiting noise. The main limitation this model is that it is not capable of capturing more fine-grained aspects of conceptual similarity, e.g. the relation between *wine*, *bottle* and *vinegar*. Capturing more fine-grained semantic relations is a major motivation for models of lexical semantics based on cross-categorization

1.3 Paper Organization

The remainder of this paper is organized into three chapters: Chapter 2 introduces the basic multi-prototype model of lexical semantics and demonstrates its sensitivity to feature noise; Chapter 3 outlines a set of Bayesian approaches based on Latent Dirichlet Allocation for automatically determining feature relevance given a set of organizational constraints; Chapter 4 details my proposed work, showing how the two models can be combined, yielding a coherent lexical semantic model capable of capturing multiple definitions of semantic relatedness guided by latent conceptual structure inferred from the data.

Chapter 2

Multi-Prototype Models of Vector-Space Lexical Semantics

Current vector-space models of lexical semantics create a single "prototype" vector to represent the meaning of a word.[1] However, due to lexical ambiguity, encoding word meaning with a single vector is problematic. This chapter outlines a method that uses clustering to produce multiple "sense-specific" vectors for each word. This approach provides a context-dependent vector representation of word meaning that naturally accommodates homonymy and polysemy. Experimental comparisons to human judgements of semantic similarity for both isolated words as well as words in sentential contexts demonstrate the superiority of this approach over both prototype and exemplar based vector-space models. Furthermore, state-of-the-art results on the WordSim-353 test collection using simple unigram features and *tf-idf* weighting are obtained. Exploring the combination of feature pruning with multi-prototype vector representations will form the basis of this thesis.

2.1 Introduction

Traditionally, word meaning is represented by a single vector of contextual features derived from co-occurrence information, and semantic similarity is computed using some measure of vector distance [48, 54]. However, due to homonymy and polysemy, capturing the semantics of a word with a single vector is problematic. For example, the word *club* is similar to both *bat* and *association,* which are not at all similar to each other. Word meaning violates the triangle inequality when viewed at the level of word types, posing a problem for vector-space models [110]. A single "prototype" vector is simply incapable of capturing phenomena such as homonymy and polysemy. Also, most vector-space models are context independent, while the meaning of a word clearly depends on context. The word *club* in "The caveman picked up the *club*" is similar to *bat* in "John hit the robber with a *bat*," but not in "The *bat* flew out of the cave."

This section describes the *multi-prototype* model, a resource-lean vector-space model that represents a word's meaning by a *set* of distinct "sense specific" vectors. The set of vectors for a word is determined by unsupervised *word sense discovery* (WSD) [98], which clusters the contexts in which a word appears. In previous work, vector-space lexical similarity and word sense discovery have been treated as two separate tasks. This proposal shows how they can be combined to create an improved vector-space model of lexical semantics. First, a word's contexts are clustered to produce groups of similar context vectors. An average "prototype" vector is then computed separately for each cluster, producing a set of vectors for each word. Finally, these cluster vectors can be used to determine the semantic similarity of both isolated words and words in context. The approach is completely

[1] The content of this chapter is derived primarily from Reisinger and Mooney [88].

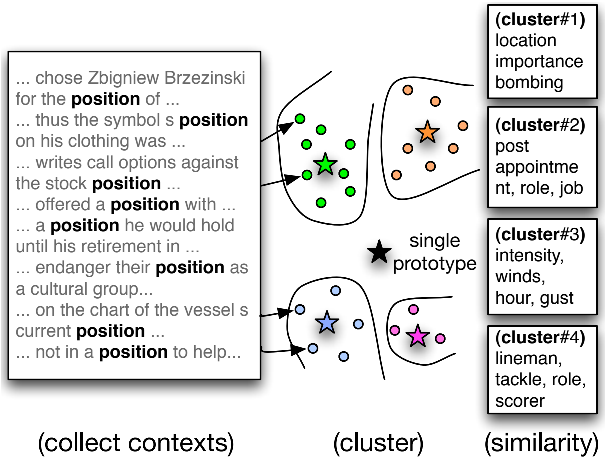

... chose Zbigniew Brzezinski for the position of ...
... thus the symbol s position on his clothing was ...
... writes call options against the stock position ...
... offered a position with ...
... a position he would hold until his retirement in ...
... endanger their position as a cultural group...
... on the chart of the vessel s current position ...
... not in a position to help...
(cluster#1) location importance bombing
(cluster#2) post appointme nt, role, job
(cluster#3) intensity, winds, hour, gust
(cluster#4) lineman, tackle, role, scorer
single prototype
(collect contexts)
(cluster)
(similarity)

Figure 2.1: Overview of the multi-prototype approach to near synonym discovery for a single target word independent of context. Occurrences are clustered and cluster centroids are used as prototype vectors. Note the "hurricane" sense of *position* (cluster 3) is not typically considered appropriate in WSD.

modular, and can integrate any clustering method with any traditional vector-space model.

The multi-prototype model is evaluated on the WordSim-353 test collection (WS-353) which consists of 353 word pairs each with 13-16 human similarity judgements [25]. When combined with aggressive feature pruning, the multi-prototype approach outperforms state-of-the-art vector space models such as Explicit Semantic Analysis [26] on WS-353, achieving rank correlation of ρ=0.77. This result rivals average human performance, obtaining correlation near that of the supervised oracle approach of Agirre et al. [2].

In addition to semantic similarity, the multi-prototype approach is also evaluated on its ability to predict the most similar words to a given target, both with and without sentential context. The results demonstrate the superiority of a clustered approach over both traditional prototype and exemplar-based vector-space models. For example, given the isolated target word *singer* the multi-prototype method produces the most similar word *vocalist*, while using a single prototype gives *musician*. Given the word *cell* in the context: "The book was published while Piasecki was still in prison, and a copy was delivered to his *cell*." the standard approach produces *protein* while the multi-prototype method yields *incarcerated*.

Finally, I demonstrate that *feature pruning* is one of the most significant factors in obtaining high correlation with human similarity judgements using vector-space models. Three approaches are evaluated: (1) basic weighted unigram collocations, (2) Explicit Semantic Analysis [ESA; 26], and (3) the *multi-prototype* model. In all three cases we show that feature pruning can be used to significantly improve correlation, in particular reaching the limit of human and oracle performance on WS-353.

2.2 Multi-Prototype Vector-Space Models

The multi-prototype model is similar to standard vector-space models of word meaning, with the addition of a per-word-type clustering step: Occurrences for a specific word type are collected from the corpus and clustered using any appropriate method (§2.2.1). Similarity between two word types is then computed as a function of their cluster centroids (§2.2.2), instead of the centroid of all the word's occurrences. Figure 2.1 gives an overview of this process.

2.2.1 Clustering Occurrences

Multiple prototypes for each word w are generated by clustering feature vectors $v(c)$ derived from each occurrence $c \in \mathcal{C}(w)$ in a large textual corpus and collecting the resulting cluster centroids $\pi_k(w), k \in [1, K]$. This approach is commonly employed in unsupervised word sense discovery; however, clusters are not intended to correspond to traditional word senses. Rather, clustering is used only to capture meaningful variation in word usage.

Our experiments employ a *mixture of von Mises-Fisher distributions* (movMF) clustering method with first-order unigram contexts [7]. Feature vectors $v(c)$ are composed of individual features $I(c, f)$, taken as all unigrams $f \in \mathcal{F}$ in a 10-word window around w.

Like spherical k-means [21], movMF models semantic relatedness using cosine similarity, a standard measure of textual similarity. However, movMF introduces an additional per-cluster *concentration* parameter controlling its semantic breadth, allowing it to more accurately model non-uniformities in the distribution of cluster sizes. Based on preliminary experiments comparing various clustering methods, movMF was found to give the best results.

2.2.2 Measuring Semantic Similarity

The similarity between two words in a multi-prototype model can be computed straightforwardly, requiring only simple modifications to standard distributional similarity methods such as those presented by Curran [19]. The similarity of isolated words can be measured using one of two *noncontextual clustered similarity metrics*:

$$\begin{aligned}
\text{AvgSim}(w, w') &\overset{\text{def}}{=} \frac{1}{K^2} \sum_{j=1}^{K} \sum_{k=1}^{K} d(\pi_k(w), \pi_j(w')) \\
\text{MaxSim}(w, w') &\overset{\text{def}}{=} \max_{1 \leqslant j \leqslant K, 1 \leqslant k \leqslant K} d(\pi_k(w), \pi_j(w'))
\end{aligned}$$

where $d(\cdot, \cdot)$ is a standard distributional similarity measure. In AvgSim, word similarity is computed as the average similarity of all pairs of prototype vectors; In MaxSim the similarity is the maximum over all pairwise prototype similarities. All results reported in this paper use *cosine* similarity, [2]

$$\text{Cos}(w, w') = \frac{\sum_{f \in \mathcal{F}} I(w, f) \cdot I(w', f)}{\sqrt{\sum_{f \in \mathcal{F}} I(w, f)^2} \sqrt{\sum_{f \in \mathcal{F}} I(w', f)^2}}$$

Both *tf-idf* feature weighting and χ^2 weighting are compared, chosen due to their ubiquity in the literature [2, 19].

In AvgSim, all prototype pairs contribute equally to the similarity computation, thus two words are judged as similar if many of their senses are similar. MaxSim, on the other hand, only requires a single pair of prototypes to be close for the words to be judged similar.

[2] The main results also hold for *weighted Jaccard* similarity.

Thus, MaxSim models the similarity of words that share only a single sense (e.g. *bat* and *club*) at the cost of lower robustness to noisy clusters that might be introduced when K is large.

When contextual information is available, AvgSim and MaxSim can be modified to produce more precise similarity computations:

$$\begin{aligned}\text{AvgSimC}(w, w') &\stackrel{\text{def}}{=} \frac{1}{K^2}\sum_{j=1}^{K}\sum_{k=1}^{K} d_{c,w,k}d_{c',w',j}d(\pi_k(w), \pi_j(w')) \\ \text{MaxSimC}(w, w') &\stackrel{\text{def}}{=} d(\hat{\pi}(w), \hat{\pi}(w'))\end{aligned}$$

where $d_{c,w,k} \stackrel{\text{def}}{=} d(v(c), \pi_k(w))$ is the likelihood of context c belonging to cluster $\pi_k(w)$, and $\hat{\pi}(w) \stackrel{\text{def}}{=} \pi_{\arg\max_{1\leqslant k\leqslant K} d_{c,w,k}}(w)$, the maximum likelihood cluster for w in context c. Thus, AvgSimC corresponds to *soft cluster assignment*, weighting each similarity term in AvgSim by the likelihood of the word contexts appearing in their respective clusters. MaxSimC corresponds to *hard assignment*, using only the most probable cluster assignment. Note that AvgSim and MaxSim can be thought of as special cases of AvgSimC and MaxSimC with uniform weight to each cluster; hence AvgSimC and MaxSimC can be used to compare words in context to isolated words as well.

2.3 Experimental Evaluation

2.3.1 Corpora

Two corpora are employed to train the models:

1. A snapshot of English Wikipedia taken on Sept. 29th, 2009. Wikitext markup is removed, as are articles with fewer than 100 words, leaving 2.8M articles with a total of 2.05B words.

2. The third edition English Gigaword corpus, with articles containing fewer than 100 words removed, leaving 6.6M articles and 3.9B words [28].

Wikipedia covers a wider range of sense distributions, whereas Gigaword contains only newswire text and tends to employ fewer senses of most ambiguous words. Our method outperforms baseline methods even on Gigaword, indicating its advantages even when the corpus covers few senses.

2.3.2 Test Collections

To evaluate the quality of various models, the automatically generated lexical similarity measurements are compared to two collections of human similarity judgements:

1. *WS-353* contains between 13 and 16 human similarity judgements for each of 353 word pairs, rated on a 1–10 integer scale [25].[3]

[3] (**Similarity vs. Relatedness**) One issue with measuring semantic similarity is that it conflates various types of relations, e.g. hyponymy, synonymy or metonymy. In order to better analyze the various components of attributional similarity, Agirre et al. [2] divide the WS-353 dataset into separate *similarity* and *relatedness* judgements. Similar pairs include synonyms, antonyms and hyponym-hypernyms; related pairs consist of meronym-holonyms and others that do not fit the previous relations. The analyses presented here are extended to these subsets.

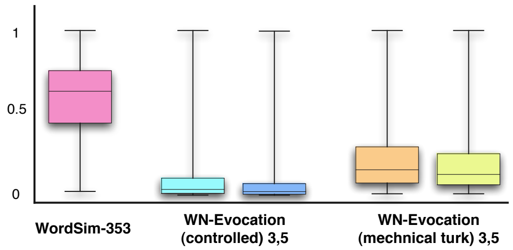


Figure 2.2: The distribution of ratings (scaled [0,1]) on WS-353 and WN-Evocation datasets. WN-Evocation consists of more low similarity pairs, even with zero-similarity pairs removed. Pairs are thresholded whether they have > 3 or > 5 ratings.

2. *WN-Evocation* contains over 100k similarity comparisons collected from both trained human raters (WN-Evocation-Controlled) and participants on Amazon's Mechanical Turk [WN-Evocation-MT; 55]. WN-Evocation comparisons are assigned to only 3-5 human raters on average and contain a significant fraction of zero- and low-similarity items compared to WS-353 (Figure 2.2). All comparisons with fewer than 3 ratings are discarded, leaving 40k pairs in WN-Evocation-Controlled with an average of 4.1 comparisons per pair and 78k pairs in WN-Evocation-MT with an average of 7.5 comparisons.

Correlation is computed using Spearman's nonparametric rank correlation (ρ) with average human judgements [2].

2.4 Judging Semantic Similarity

Figure 2.3 plots Spearman's ρ on WordSim-353 against the number of clusters (K) for Wikipedia and Gigaword corpora, using pruned *tf-idf* and χ^2 features.[4] In general pruned *tf-idf* features yield higher correlation than χ^2 features. Using AvgSim, the multi-prototype approach ($K > 1$) yields higher correlation than the single-prototype approach ($K = 1$) across all corpora and feature types, achieving state-of-the-art results ($\rho = 0.77$) with pruned *tf-idf* features. This result is statistically significant in all cases for *tf-idf* and for $K \in [2, 10]$ on Wikipedia and $K > 4$ on Gigaword for χ^2 features.[5] MaxSim yields similar performance when $K < 10$ but performance degrades as K increases.

It is possible to circumvent the model-selection problem (choosing the best value of K) by simply combining the prototypes from clusterings of different sizes. This approach represents words using both semantically broad and semantically tight prototypes, similar to hierarchical clustering. Table 2.1 and Figure 2.3 (squares) show the result of such a *combined* approach, where the prototypes for clusterings of size 2-5, 10, 20, 50, and 100 are unioned to form a single large prototype set. In general, this approach works about as well as picking the optimal value of K, even outperforming the single best cluster size for Wikipedia.

[4] (**Feature pruning**) Results using *tf-idf* features are extremely sensitive to feature pruning while χ^2 features are more robust. In all experiments *tf-idf* features are pruned by their overall weight, taking the top 5000. This setting was found to optimize the performance of the single-prototype approach.

[5] Significance is calculated using the large-sample approximation of the Spearman rank *test*; ($p < 0.05$).

Spearman's ρ	**prototype**	**exemplar**	**multi-prototype (AvgSim)**			**combined**
			$K = 5$	$K = 20$	$K = 50$	
Wikipedia *tf-idf*	0.53±0.02	0.60±0.06	0.69±0.02	0.76±0.01	0.76±0.01	0.77±0.01
Wikipedia χ^2	0.54±0.03	0.65±0.07	0.58±0.02	0.56±0.02	0.52±0.03	0.59±0.04
Gigaword *tf-idf*	0.49±0.02	0.48±0.10	0.64±0.02	0.61±0.02	0.61±0.02	0.62±0.02
Gigaword χ^2	0.25±0.03	0.41±0.14	0.32±0.03	0.35±0.03	0.33±0.03	0.34±0.03

Table 2.1: Spearman correlation on the WordSim-353 dataset broken down by corpus and feature type.

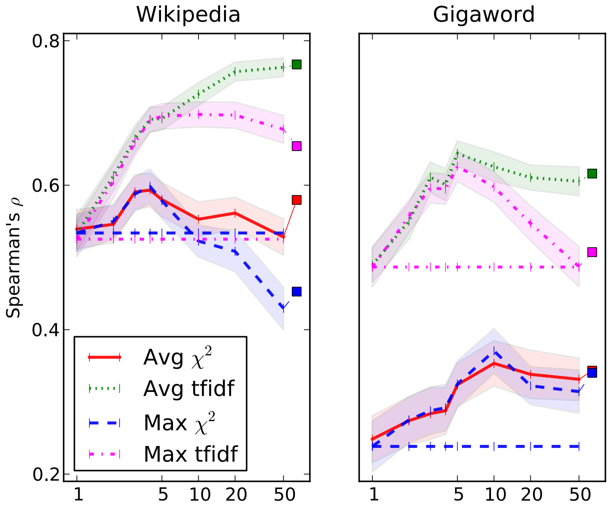


Figure 2.3: WordSim-353 rank correlation vs. number of clusters (log scale) using AvgSim and MaxSim on both the Wikipedia (left) and Gigaword (right) corpora. Horizontal bars show the performance of single-prototype. Squares indicate performance when combining across clusterings. Error bars depict 95% confidence intervals using the Spearman test. Squares indicate performance when combining across clusterings.

Finally, the multi-prototype approach is also compared to a pure exemplar approach, averaging similarity across all occurrence pairs.[6] Table 2.1 summarizes the results. The exemplar approach yields significantly higher correlation than the single prototype approach in all cases except Gigaword with *tf-idf* features ($p < 0.05$). Furthermore, it performs significantly *worse* than combined multi-prototype for *tf-idf* features, and does not differ significantly for χ^2 features. Overall this result indicates that multi-prototype performs at least as well as exemplar in the worst case, and significantly outperforms when using the best feature representation / corpus pair.

2.5 Predicting Near-Synonyms

The multi-prototype model is next evaluated on on its ability to determine the most closely related words for a given target word (using the Wikipedia corpus with *tf-idf* features).

[6] Averaging across all pairs was found to yield higher correlation than averaging over the most similar pairs.

homonymous
carrier, crane, cell, company, issue, interest, match, media, nature, party, practice, plant, racket, recess, reservation, rock, space, value
polysemous
cause, chance, journal, market, network, policy, power, production, series, trading, train

Table 2.2: Words used in predicting near synonyms.

The top k most similar words were computed for each prototype of each target word. Using a forced-choice setup, human subjects were asked to evaluate the quality of these *near synonyms* relative to those produced by a single prototype. Participants on Amazon's Mechanical Turk[7] [103] were asked to choose between two possible alternatives (one from a prototype model and one from a multi-prototype model) as being most similar to a given target word. The target words were presented either in isolation or in a sentential context randomly selected from the corpus. Table 2.2 lists the ambiguous words used for this task. They are grouped into homonyms (words with very distinct senses) and polysemes (words with related senses). All words were chosen such that their usages occur within the same part of speech.

In the non-contextual task, 79 unique raters completed 7,620 comparisons of which 72 were discarded due to poor performance on a known test set.[8] In the contextual task, 127 raters completed 9,930 comparisons of which 87 were discarded.

For the non-contextual case, Figure 2.4 left plots the fraction of raters preferring the multi-prototype prediction (using AvgSim) over that of a single prototype as the number of clusters is varied. When asked to choose between the single best word for each method (**top word**), the multi-prototype prediction is chosen significantly more frequently (i.e. the result is above 0.5) when the number of clusters is small, but the two methods perform similarly for larger numbers of clusters (Wald test, $\alpha = 0.05$.) Clustering more accurately identifies homonyms' clearly distinct senses and produces prototypes that better capture the different uses of these words. As a result, compared to using a single prototype, the multi-prototype approach produces better near synonyms for homonyms compared to polysemes. However, given the right number of clusters, it also produces better results for polysemous words.

The near synonym prediction task highlights one of the weaknesses of the multi-prototype approach: as the number of clusters increases, the number of occurrences assigned to each cluster decreases, increasing noise and resulting in some poor prototypes that mainly cover outliers. The word similarity task is somewhat robust to this phenomenon, but synonym prediction is more affected since only the top predicted choice is used. When raters are forced to chose between the top *three* predictions for each method (presented as **top set** in Figure 2.4 left), the effect of this noise is reduced and the multi-prototype approach remains dominant even for a large number of clusters. This indicates that although more clusters can capture finer-grained sense distinctions, they also can introduce noise.

[7] http://mturk.com

[8] (**Rater reliability**) The reliability of Mechanical Turk raters is quite variable, so rater quality was evaluated by including control questions with a known correct answers in each HIT. Control questions were generated by selecting a random word from WordNet 3.0 and including as possible choices a word in the same synset (correct answer) and a word in a synset with a high path distance (incorrect answer). Raters who got less than 50% of these control questions correct, or spent too little time on the HIT were discarded.

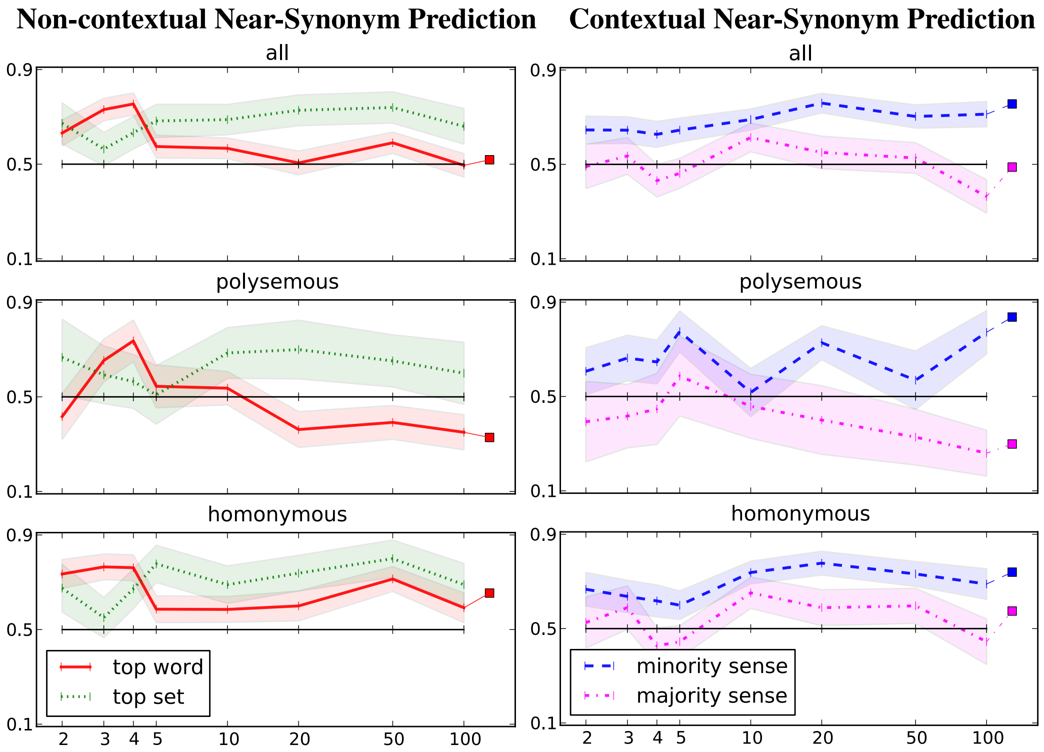


Figure 2.4: (**left**) Near-synonym evaluation for isolated words showing fraction of raters preferring multi-prototype results vs. number of clusters. Colored squares indicate performance when combining across clusterings. 95% confidence intervals computed using the Wald test. (**right**) Near-synonym evaluation for words in a sentential context chosen either from the minority sense or the majority sense.

When presented with words in context (Figure 2.4 right),[9] raters found no significant difference in the two methods for words used in their majority sense.[10] However, when a minority sense is presented (e.g. the "prison" sense of *cell*), raters prefer the choice predicted by the multi-prototype approach. This result is to be expected since the single prototype mainly reflects the majority sense, preventing it from predicting appropriate synonyms for a minority sense. Also, once again, the performance of the multi-prototype approach is better for homonyms than polysemes.

2.6 Effects of Pruning

In addition to feature weighting, adequate pruning of irrelevant features is critical when computing semantic relatedness. This section demonstrates basic results using a simple *fixed window* pruning scheme [26], keeping a fixed number of features (ordered by weight) for each term. Table 2.3 summarizes the results. Several different feature weighting are evaluated: *tf*, *tf-idf*, *t-test*, and χ^2 [20]. Feature vectors are pruned to a fixed length f,

[9] Results for the multi-prototype method are generated using AvgSimC (soft assignment) as this was found to significantly outperform MaxSimC.

[10] Sense frequency determined using Google; senses labeled manually by trained human evaluators.

Method	**WordSim-353**			**WN-Evocation**	
	Sim.	Rel.	Both	Controlled	Turk
Human[a]	0.78	0.74	0.75	0.02	0.37
Agirre et al. [2]					
best unsup.[b]	0.72	0.56	0.66	-	-
best oracle[c]	**0.83**	0.71	**0.78**	-	-
Single Prototype					
all	0.26	0.29	0.25	0.10	0.10
$f = 1000$	0.76	0.72	0.73	0.21	0.16
$f = 5000$	0.65	0.55	0.59	0.15	0.13
$f = 10000$	0.56	0.46	0.52	0.14	0.12
Multi-Prototype (50 clusters)[d]					
all	0.07	0.17	0.07	0.05	0.08
$f^* = 1000$	0.78	0.70	0.74	**0.25**	**0.16**
$f^* = 5000$	**0.81**	**0.76**	**0.77**	0.24	0.16
$f^* = 10000$	0.79	0.74	0.74	0.24	0.15
Explicit Semantic Analysis					
all	0.58	0.59	0.56	-	-
$f = 1000$	0.75	0.66	0.70	-	-
$f = 5000$	0.77	0.73	0.74	-	-
$f = 10000$	0.77	0.74	0.74	-	-

[a] Surrogate human performance computed using leave-one-out Spearman's ρ averaged across raters for WS-353 and randomized for WN-Evocation. In WN-Evocation, the small number of ratings per pair and randomization makes LOO an unreliable estimator and thus should be interpreted as a rough lower bound.

[b] WordNet-based multilingual approach.

[c] Supervised combination of b, context-window features and syntactic features.

[d] Effective number of features, $f^* \overset{\text{def}}{=} f/K$ is given in order to enforce a fair comparison.

Table 2.3: Correlation results on WS-353 and WN-Evocation comparing previous studies and surrogate human performance to weighted unigram collocations with feature pruning. Prototype and ESA-based approaches shown use *tf-idf* weighting and cosine distance. Multi-prototype results are given for 50 clusters ($K = 50$).

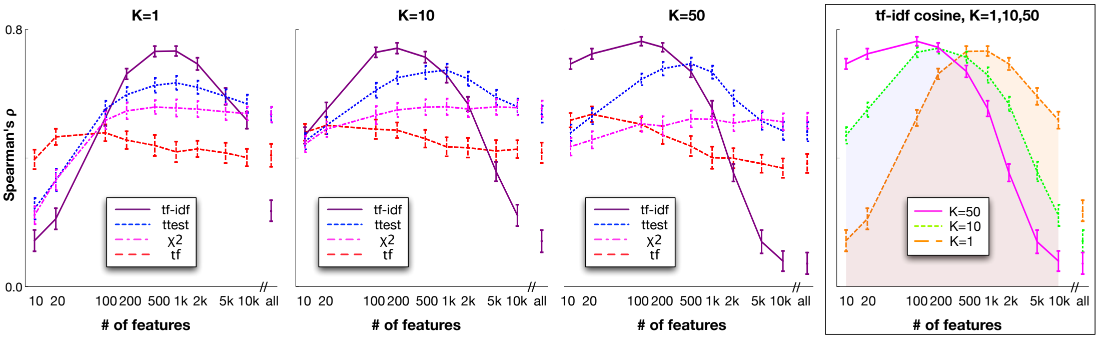


Figure 2.5: Effects of feature pruning and representation on WS-353 correlation broken down across multi-prototype representation size. In general *tf-idf* features are the most sensitive to pruning level, yielding the highest correlation for moderate levels of pruning and significantly lower correlation than other representations without pruning. The optimal amount of pruning varies with the number of prototypes used, with fewer features being optimal for more clusters.

discarding all but the highest-weight features.

For WS-353, unigram collocations perform the worst without pruning (ρ=0.25 for multi-prototype and ρ=0.25 for single prototype), followed by ESA (ρ=0.59), but that with optimal pruning both methods perform about the same (ρ=0.73 and ρ=0.74 respectively). The unpruned multi-prototype approach does poorly with *tf-idf* features because it amplifies feature noise by partitioning the raw occurrences. When employing feature pruning, however, unigram collocations outperform ESA and across a wide range of pruning levels. Note that pruning clearly helps in all three test cases and across a wide range of settings for f (cf. Figure 2.5 and Figure 2.6).

For WN-Evocation, there is significant benefit to feature pruning in both the single-prototype and multi-prototype case. The best correlation results are again obtained using pruned *tf-idf* with multiple-prototypes (ρ=0.25 for controlled and ρ=0.16 for Mechanical Turk), although *t-test* features also perform well and benefit from pruning.

The optimal pruning cutoff depends on the feature weighting and number of prototypes (Figure 2.5) as well as the feature representation (Figure 2.6). *t-test* and χ^2 features are most robust to feature noise and perform well even with no pruning; *tf-idf* yields the best results but is sensitive to the pruning parameter. As the number of increases, more pruning is required to combat feature noise.

Figure 2.6 breaks down the similarity pairs into four quantiles for each data set and then shows correlation separately for each quantile. In general the more polarized data quantiles (1 and 4) have higher correlation, indicating that fine-grained distinctions in semantic distance are easier for those sets. The fact that the per-quantile correlation is significantly lower than the full correlation e.g. in the human case means that fine-grained ordering (within quantile) is more difficult than coarse-grained (between quantile). Feature pruning improves correlations in quantiles 2–4 while reducing correlation in quantile 1. This result is to be expected as more features are necessary to make fine-grained distinctions between dissimilar pairs.

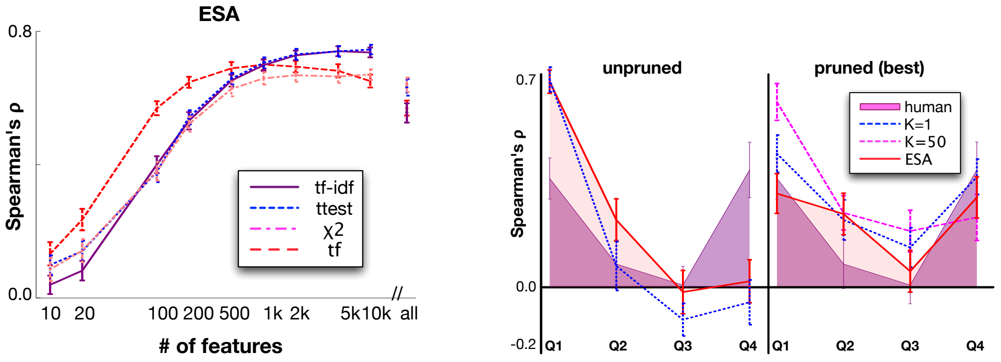


Figure 2.6: (**left**) Effects of feature pruning using ESA on WS-353; more features are required to attain high correlation compared to unigram collocations. (**right**) Correlation results on WS-353 broken down over quantiles in the human ratings. Quantile ranges are shown in Figure 2.2. In general ratings for highly similar (dissimilar) pairs are more predictable (quantiles 1 and 4) than middle similarity pairs (quantiles 2, 3).

2.7 Discussion

This chapter introduced a resource-light model for vector-space word meaning that represents words as collections of prototype vectors, naturally accounting for lexical ambiguity. This multi-prototype approach uses word sense discovery to partition a word's contexts and construct "sense specific" prototypes for each cluster. Doing so significantly increases the accuracy of lexical-similarity computation as demonstrated by improved correlation with human similarity judgements and generation of better near synonyms according to human evaluators. Furthermore, although performance is sensitive to the number of prototypes, combining prototypes across a large range of clusterings performs nearly as well as the ex-post best clustering.

Compared to WordNet, the best-performing clusterings are significantly more fine-grained. Furthermore, they often do not correspond to agreed upon semantic distinctions (e.g., the "hurricane" sense of *position* in Fig. 2.1). The finer-grained senses are posited to actually capture useful aspects of word meaning, leading to better correlation with WordSim-353. In proposed work I will explore multiple simultaneous clusterings, for discovering even more fine-grained sense distinctions. Although these senses may not be useful from a lexicographical standpoint, I posit that they nevertheless do capture important semantic information.

Finally, I have demonstrated that feature pruning for distributional similarity can significantly improve correlation with human similarity and relatedness judgements. Feature selection combined with the multi-prototype representation achieves state-of-the-art results on the WordSim-353 task, beating a measure of human performance, and performing nearly as well as a supervised oracle approach. The complexity of the interaction between feature weighting and pruning and magnitude of their combined effect on correlation strongly suggests that they should be studied in greater detail, and form a major component of my proposed work: The next chapter discusses model-based strategies for feature weighting and noise reduction in the context of attribute extraction.

CHAPTER 3

Determining Feature Relevance with Topic Models

3.1 Introduction

This chapter describes a set of Bayesian methods for automatically filtering noisy features in the context of *attribute extraction*, i.e. automatically determining the cognitively salient attributes for a given set of concepts.[1] Examples of attributes include "height" and "eye-color" for the concept *Person* or "GDP" and "president" for *Country*. Identifying and extracting such attributes relative to a set of flat (i.e., non-hierarchically organized) labeled classes of instances has been extensively studied, using a variety of data, e.g., Web search query logs [67], Web documents [122], and Wikipedia [105, 119]. Attributes are useful in a variety of applications, e.g., question answering [108], unsupervised semantic parsing [82], query intent classification and expansion [41], automatic infobox generation for Wikipedia entries [120], and associative anaphora resolution [13, 97].

Attributes are extracted for a set of pre-defined concepts following the open-domain distributional methods developed by Paşca and Van Durme [67]. Contextual features are first collected for candidate attributes and then candidate attributes from other classes are ranked according to their distributional similarity to a small seed set provided for a single known semantic class. Post-extraction noise filtering and attribute re-ranking is accomplished leveraging a set of generative models imposing a variety of structural constraints. In particular I study the effects of hierarchical constraints both *explicitly* encoded in the concept-graph structure of WordNet, and *implicitly* encoded by *hierarchical latent class models* such as the Nested Chinese Restaurant Process [10]. Various assumptions about feature generalization in the latent concept space are implemented and tested using models based on Latent Dirichlet Allocation [11]. These models reweight individual attributes based on their likelihood given the model assumptions, and hence can be used for smoothing or attribute filtering.

Three models were compared: (1) Flat (unstructured) groupings of properties inferred using Latent Dirichlet Allocation; (2) Fixing the concept structure to conform to WordNet 3.0 categories; (3) Inferring the concept structure automatically assuming a latent hierarchy constraint based on the nested Chinese Restaurant Process [10]. The structured models are capable of inferring per-feature concept specificity, allowing them to capture finer-grained concept structure (e.g. *paintings* is more specific an attribute for *painter* than *height*). Also the discriminative power of attributes/features is captured directly in the hierarchical structure, with the least discriminative attributes at the root. This model allows the extraction of very specific attributes (e.g. that the Rosybill is *endemic to South America*) while reducing noise overall.

[1] The content of this chapter is largely derived from Reisinger and Paşca [89].

anticancer drugs: mechanism of action, uses, extravasation, solubility, contraindications, side effects, chemistry, molecular weight, history, mode of action

bollywood actors: biography, filmography, age, biodata, height, profile, autobiography, new wallpapers, latest photos, family pictures

citrus fruits: nutrition, health benefits, nutritional value, nutritional information, calories, nutrition facts, history

european countries: population, flag, climate, president, economy, geography, currency, population density, topography, vegetation, religion, natural resources

london boroughs: population, taxis, local newspapers, mp, lb, street map, renault connexions, local history

microorganisms: cell structure, taxonomy, life cycle, reproduction, colony morphology, scientific name, virulence factors, gram stain, clipart

renaissance painters: early life, bibliography, short biography, the david, bio, painting, techniques, homosexuality, birthplace, anatomical drawings, famous paintings

Figure 3.1: Examples of labeled attribute sets extracted using the method from [67].

Previous work has maintained the assertion that smoothing directly conflicts with the task of attribute extraction as psychological saliency is often tied to the feature's discriminative power, and hence rare features are deemed more salient [8], i.e. "is ferocious" is a more salient feature for *Lion* than "is an animal." However such overly specific features are often discarded by smoothing. In this work we show a more complex relationship between smoothing and salient property identification: while standard "flat" smoothing methods do indeed promote more generic properties over salient ones, we find that hierarchical smoothing is capable of separating generic and specific features, weighting discriminative properties more highly.

The resulting models are evaluated along two dimensions: (1) the precision of the ranked lists of attributes, and (2) the quality of the attribute assignments to WORDNET concepts. In all cases we find that the principled LDA-based approaches outperform previously proposed heuristic methods, greatly improving the specificity of attributes at each concept.

3.2 Attribute Extraction

Input to the model-based smoothing procedure consists of sets of class instances (e.g., Pisanello, Hieronymus Bosch) associated with class labels (e.g., *renaissance painters*) and attributes (e.g., "birthplace", "famous works", "style" and "early life"; Table 3.1). Clusters of noun phrases (instances) are constructed using distributional similarity [50, 36] and are labeled by applying "such-as" surface patterns to raw Web text (e.g., "*renaissance painters* such as Hieronymous Bosch"), yielding 870K instances in more than 4500 classes [67].

Attributes for each flat labeled class are extracted from anonymized Web search query logs using the minimally supervised procedure in [65][2]. Candidate attributes are ranked based on their weighted Jaccard similarity to a set of 5 manually provided seed attributes for the class *european countries*. Figure 3.1 illustrates several such *labeled attribute sets* (the underlying instances are not depicted). Naturally, the attributes extracted are not perfect, e.g., "lb" and "renault connexions" as attributes for *london boroughs*.

[2] Similar query data, including query strings and frequency counts, is available from, e.g., [27]

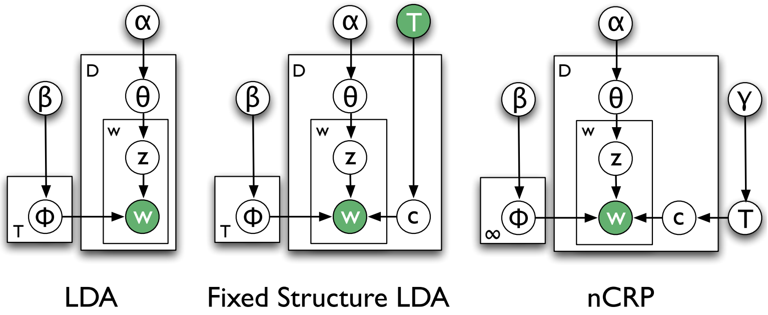


Figure 3.2: Graphical models for the LDA feature re-ranking models.

3.3 Topic Models for Feature Weighting

This section details application of the LDA-based attribute smoothing/re-ranking models (§3.3.1-§3.3.3) to the *Labeled attribute sets* generated using the above extraction procedure, and an approach for querying the predictive density of the inferred models to get robust attribute rankings (§3.3.4).

3.3.1 Latent Dirichlet Allocation

The underlying model for attribute re-ranking is LDA [11], a fully Bayesian extension of probabilistic Latent Semantic Analysis [39]. Given D labeled attribute sets $\mathbf{w}_d$, $d \in D$, LDA infers an unstructured set of T latent *annotated concepts* over which attribute sets decompose as mixtures.[3] The latent annotated concepts represent semantically coherent groups of attributes expressed in the data, as shown in Example 1. We choose LDA over the conceptually simpler LSA as it can be more easily extended to account for hierarchical latent structure, a fact we will take advantage of here. Furthermore, unlike simply clustering attribute sets, LDA decomposes attributes into latent "topics" that can be multiply inherited: e.g. the category *Lion* could inherit attributes from both from the *organism* and *big-cat* topic, leading to more semantically pure smoothing.

In topic models such as LDA, documents are drawn from a weighted average over a set of latent *topics* T. Each document $\mathbf{w}_d$ maintains a separate multinomial distribution $\boldsymbol{\theta}_d$ over topics $\boldsymbol{\phi}$. For each word $w_{i,d}$ a *topic index* $z_{i,d}$ is drawn from $\boldsymbol{\theta}_d$ and then $w_{i,d}$ is drawn from the corresponding topic multinomial $\boldsymbol{\phi}_{z_{i,d}}$. The generative model is given by

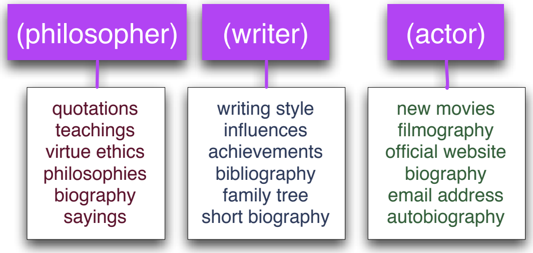
$$\begin{array}{rcll}
\boldsymbol{\theta}_d|\boldsymbol{\alpha} & \sim & \text{Dirichlet}(\boldsymbol{\alpha}), \quad d \in D, & \text{(topic prop.)} \\
\boldsymbol{\phi}_t|\boldsymbol{\beta} & \sim & \text{Dirichlet}(\boldsymbol{\beta}), \quad t \in T, & \text{(topics)} \\
z_{id}|\boldsymbol{\theta}_d & \sim & \text{Mult}(\boldsymbol{\theta}_d), \quad i \in |\mathbf{w}_d|, & \text{(topic ind.)} \\
w_{id}|\boldsymbol{\phi}_{z_{id}} & \sim & \text{Mult}(\boldsymbol{\phi}_{z_{id}}), \quad i \in |\mathbf{w}_d|, & \text{(words)}
\end{array}$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are hyperparameters smoothing the per-document topic distributions and per-topic word distributions respectively.

We are interested in the case where $\mathbf{w}$ is known and we want to compute the conditional posterior of the remaining random variables $p(\mathbf{z}, \boldsymbol{\beta}, \boldsymbol{\theta}|\mathbf{w})$. This distribution can be approximated efficiently using Gibbs sampling. See [11] and [31] for more details.

[3] In topic modeling literature, attributes are *words* and attribute sets are *documents*.

(**Example 1**) Given 26 labeled attribute sets falling into three broad semantic categories: philosophers, writers and actors (e.g., attribute sets for *contemporary philosophers*, *women writers*, and *bollywood actors*), LDA is able to infer a meaningful set of latent annotated concepts without any knowledge of the actual input category labels:



(labels manually added for the latent annotated concepts are shown in parentheses). Note that with a flat concept structure, attributes can only be separated into broad clusters, so the generality/specificity of attributes cannot be inferred. Parameters were α=1, η=0.1, T=3.

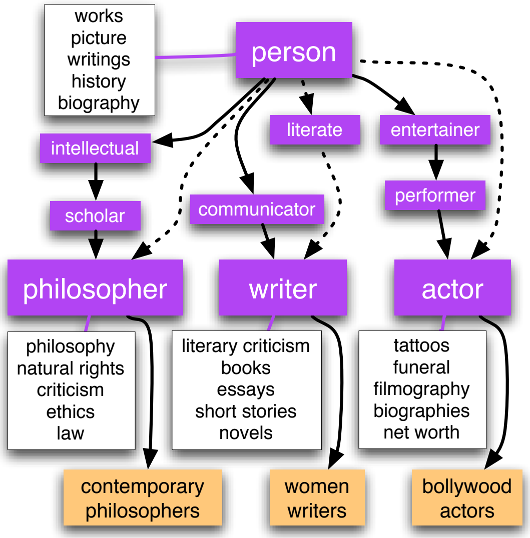
3.3.2 Labeled LDA

LDA can be extended to model structural dependencies between latent annotated concepts (cf. [49, 102]); In particular, we fix the concept structure to correspond to the WORDNET Is-A hierarchy using *Labeled* LDA [83]. In Labeled LDA the document to topic mapping is fixed and specified *a priori*. This model allows rich categorical structure to be incorporated into LDA without increasing its computational complexity.

Labeled LDA maintains a vector of topic-proportions $\boldsymbol{\theta}_d$ for each document as in LDA, as well a fixed binary topic absence/presence vector mask $\boldsymbol{\Lambda}_d = (l_1, l_2, \cdots, l_t)$, $l_t \in 0, 1$, indicating whether each topic is considered to be related to that document. By projecting $\mathbf{w}_d$ only onto the subset of topics indicated by $\boldsymbol{\Lambda}_d$, the model is forced to generalize in a way that is consistent with the overlapping topic structure. For example, with a tree structure, $\mathbf{c}_d$ would be constrained to correspond to the concept nodes in T on the path from the root to the leaf containing d. In the case of WORDNET, we can fix $\boldsymbol{\Lambda}_d$ to correspond to the set of nodes from d to *WN_entity* following the Is-A edges. Furthermore, we can easily combine Labeled LDA with flat LDA, resulting in a hybrid model capable of accounting for explicit *and* latent structure.

In order leverage the structure of WORDNET in Labeled LDA, each labeled attribute set is assigned to a leaf concept in WORDNET based on the edit distance between the WORDNET concept label and the attribute set label. Possible latent concepts for this set include the concepts along all paths from its attachment point to the WORDNET root, following Is-A relation edges. Therefore, any two labeled attribute sets share a number of latent concepts based on their similarity in WORDNET: all labeled attribute sets share at least the root concept, and may share more concepts depending on their most specific, common ancestor. Under such a model, more general attributes naturally attach to latent concept nodes closer to the root, and more specific attributes attach lower (Example 2). The WORDNET-based model is referred to as *Fixed Structure LDA* (fsLDA).

(**Example 2**) Fixing the latent concept structure to correspond to WORDNET (dark/purple nodes), and attaching each labeled attribute set (examples depicted by light/orange nodes) yields the annotated hierarchy:



Attribute distributions for the small nodes are not shown. Unlike with the flat annotated concept structure, with a hierarchical concept structure, attributes can be separated by their generality. Parameters were set at $\alpha{=}1$ and $\eta{=}0.1$.

Example 2 shows the property distributions inferred when fixing each topic to correspond to a WORDNET category using fsLDA. At a high level the topics inferred by LDA correspond roughly to the main categories (Example 1; *Actors*, *Philosophers* and *Writers*), but fsLDA is better able to capture more fine-grained structure, separating out the attributes common to all input attribute sets in the root *Person* node.

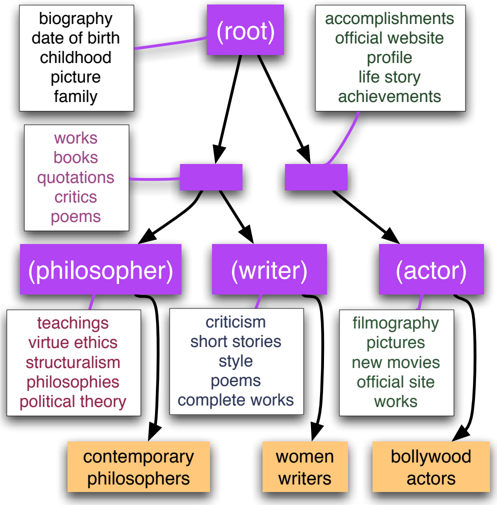
3.3.3 Hierarchical LDA

By putting a prior distribution over $\mathbf{\Lambda}$, it is possible to extend Labeled LDA to automatically infer which subsets of topics a particular document might have affinity. Blei et al. [10] describe one approach to such *hierarchical* LDA (hLDA) setting $\mathbf{\Lambda}$ to correspond to a random fixed-depth tree structure with infinite branching factor, leveraging the *nested Chinese Restaurant Process* (nCRP) prior. Unlike the fixed-structure approach which uses the WORDNET hierarchy directly, the nCRP generates its own annotated hierarchy to explain the data (Example 3).

Operationally, Labeled LDA is extended with $\mathbf{\Lambda}_d|\gamma \sim \text{nCRP}(\gamma, L)$, $d \in D$, where γ is a hyperparameter controlling the probability of branching via a per-node Dirichlet Process, and L is the fixed tree depth. Each document d maintains a depth L path to the root; all documents share at least one topic (the root) and possibly more depending on their similarity. The resulting model infers a hierarchy of topics describing the data, penalizing attribute generalization that does not conform to the strict tree structure.

From a concept organization perspective, each node in this model corresponds to a latent concept with an arbitrary number of subconcepts. Furthermore, hierarchical LDA differs from the previous two models in that it can enforce a strict notion of hierarchy, penalizing attributes that violate the inferred tree structure.

(**Example 3**) Applying hLDA to the same three semantic categories: philosophers, writers and actors, yields the model:



(manually added labels are shown in parentheses). Unlike in WORDNET, the inferred structure naturally places philosopher and writer under the same subconcept, which is also separate from actor. Hyperparameters were α=0.1, η=0.1, γ=1.0.

hLDA is able to infer a latent organizational structure directly from the data which can differ substantially from fsLDA, e.g. placing *Philosophers* and *Writers* under the same subnode separate from *Actors*, unifying them based on their shared attributes related to writing.

3.3.4 Model-based Property Ranking

Inferred models can be queried to provide ranked attribute lists in three ways:

Per-Node Distribution: In fsLDA, attribute rankings can be constructed directly for each WORDNET concept c, by computing the likelihood of attribute w attaching to c, $\mathcal{L}(c|w) = p(w|c)$ averaged over all Gibbs samples (discarding a fixed number of samples for burn-in). Since c's attribute distribution is not dependent on the distributions of its children, the resulting distribution is biased towards more specific attributes.

Class-Entropy (CE): In all models, the inferred latent annotated concepts can be used to *smooth* the attribute rankings for each labeled attribute set. Each sample from the posterior is composed of two components: (1) a multinomial distribution over a set of WORDNET nodes, $p(c|\mathbf{w}_d, \alpha)$ for each attribute set $\mathbf{w}_d$, where the (discrete) values of c are WORDNET concepts, and (2) a multinomial distribution over attributes $p(w|c, \eta)$ for each WORDNET concept c. To compute an attribute ranking for $\mathbf{w}_d$, we have

$$p(w|\mathbf{w}_d) = \sum_c p(w|c,\eta)p(c|\mathbf{w}_d,\alpha).$$

Given this new ranking for each attribute set, we can compute new rankings for each WordNet concept c by averaging again over all the $\mathbf{w}_d$ that appear as (possible indirect) descendants of c. Thus, this method uses LDA to first perform reranking on the raw extractions before applying the baseline ontology induction procedure (§ 3.4.2).[4]

CE ranking exhibits a "conservation of entropy" effect, whereby the proportion of general to specific attributes in each attribute set $\mathbf{w}_d$ remains the same in the posterior. If set A contains 10 specific attributes and 30 generic ones, then the latter will be favored over the former in the resulting distribution 3 to 1. Conservation of entropy is a strong assumption, and in particular it hinders improving the *specificity* of attribute rankings.

Class-Entropy+Prior: The LDA-based models do not inherently make use of any ranking information contained in the original extractions. However, such information can be incorporated in the form of a prior. The final ranking method combines CE with an exponential prior over the attribute rank in the baseline extraction. For each attribute set, we compute the probability of each attribute

$$p(w|\mathbf{w}_d) = p_{\text{lda}}(w|\mathbf{w}_d)p_{\text{base}}(w|\mathbf{w}_d),$$

assuming a parametric form for $p_{\text{base}}(w|\mathbf{w}_d) \overset{\text{def}}{=} \theta^{r(w,\mathbf{w}_d)}$. Here, $r(w,\mathbf{w}_d)$ is the rank of w in attribute set d. In all experiments reported, θ=0.9.

3.4 Experimental Setup

3.4.1 Data Analysis

Input labeled attribute sets are derived using the procedure in § 3.2 There are 4502 input attribute sets with a total of 225K attributes (24K unique), of which 8121 occur only once. The 10 attributes occurring in the most sets (history, definition, picture(s), images, photos, clipart, timeline, clip art, types) account for 6% of the total.

3.4.2 Model Settings

Baseline: Each labeled attribute set is mapped to the most common WordNet concept with the closest label string distance [65]. Attributes are propagated up the tree, attaching to node c if they are contained in a majority of c's children.

LDA: LDA is used to infer a flat set of $T = 300$ latent annotated concepts describing the data. The concept selection smoothing parameter is set as α=100. The smoother for the per-concept multinomial over words is set as η=0.1.[5] The effects of concept structure on attribute precision can be isolated by comparing the structured models to LDA.

Fixed-Structure LDA (fsLDA): The latent concept hierarchy is fixed based on WordNet (§ 3.3.2), and labeled attribute sets are mapped into it as in *baseline*. The concept graph for

[4] One simple extension is to run LDA again on the CE ranked output, yielding an iterative procedure; however, this was not found to significantly affect precision.

[5] **(Parameter setting)** Across all models, the main results in this paper are robust to changes in α. For hLDA, changes in η and γ affect the size of the learned model but have less effect on the final precision. Larger values for L give the model more flexibility, but take longer to train.

each labeled attribute set $\mathbf{w}_d$ is decomposed into (possibly overlapping) chains, one for each unique path from the WORDNET root to $\mathbf{w}_d$'s attachment point. Each path is assigned a copy $\mathbf{w}_d$, reducing the bias in attribute sets with many unique ancestor concepts.[6] The final models contain 6566 annotated concepts on average.

Arbitrary hierarchy (hLDA): For the arbitrary hierarchy model (§3.3.3), we set the maximum tree depth L=5, per-concept attribute smoother η=0.05, concept assignment smoother α=10 and hLDA branching proportion γ=1.0. The resulting models span 380 annotated concepts on average.

3.4.3 Evaluating Attribute Attachment

For the WORDNET-based models, in addition to measuring the average precision of the reranked attributes, it is also useful to evaluate the assignment of attributes to WORDNET concepts. For this evaluation, human annotators were asked to determine the most appropriate WORDNET synset(s) for a set of gold attributes, taking into account polysemous usage. For each model, ranked lists of possible concept assignments $C(w)$ are generated for each attribute w, using $\mathcal{L}(c|w)$ for ranking. The accuracy of a list $C(w)$ for an attribute w is measured by a scoring metric that corresponds to a modification [66] of the mean reciprocal rank score [114]:

$$DRR = \max \frac{1}{rank(c) \times (1 + PathToGold)}$$

where $rank(c)$ is the rank (from 1 up to 10) of a concept c in $C(w)$, and PathToGold is the length of the minimum path along Is-A edges in the conceptual hierarchies between the concept c, on one hand, and any of the gold-standard concepts manually identified for the attribute w, on the other hand. The length PathToGold is 0, if the returned concept is the same as the gold-standard concept. Conversely, a gold-standard attribute receives no credit (that is, DRR is 0) if no path is found in the hierarchies between the top 10 concepts of $C(w)$ and any of the gold-standard concepts, or if $C(w)$ is empty. The overall precision of a given model is the average of the DRR scores of individual attributes, computed over the gold assignment set [66].

3.5 Results

3.5.1 Attribute Precision

Precision was manually evaluated relative to 23 concepts chosen for broad coverage.[7] Table 3.1 shows precision at n and the Mean Average Precision (MAP); In all LDA-based models, the Bayes average posterior is taken over all Gibbs samples after burn-in.[8] The improvements in average precision are important, given the amount of noise in the raw extracted data.

[6] Reducing the directed-acyclic graph to a tree ontology did not significantly affect precision.

[7] **(Precision evaluation)** Attributes were hand annotated using the procedure in [67] and numerical precision scores (1.0 for vital, 0.5 for okay and 0.0 for incorrect) were assigned for the top 50 attributes per concept. 25 reference concepts were originally chosen, but 2 were not populated with attributes in any method, and hence were excluded from the comparison.

[8] **(Bayes average vs. maximum a-posteriori)** The full Bayesian average posterior consistently yielded higher precision than the maximum a-posteriori model. For the per-node distributions, the fsLDA Bayes average model exhibits a 17% reduction in relative error over the maximum a-posteriori estimate.

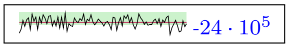
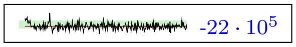
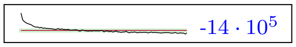
Model	Precision @				MAP	DRR			
	5	10	20	50		all	(n)	found	(n)
Base (unranked)	0.45	0.48	0.47	0.44	0.46	0.14	(150)	0.24	(91)
Base (ranked)	0.77	0.77	0.69	0.58	0.67	0.17	(150)	0.21	(123)
LDA†									
CE	0.64	0.53	0.52	0.56	0.55				
CE+Prior	0.80	0.73	0.74	0.58	0.69				
Fixed-structure (fsLDA)									
Per-Node	0.43	0.41	0.42	0.41	0.42	0.31	(150)	0.37	(128)
CE	0.75	0.68	0.63	0.55	0.63				
CE+Prior	0.78	0.77	0.71	0.59	0.69				
hLDA†									
CE	0.74	0.76	0.73	0.65	0.72				
CE+Prior	0.88	0.85	0.81	0.68	0.78				

Table 3.1: Precision at n and mean-average precision for all models and data sets. Inset plots show log-likelihood of each Gibbs sample, indicating convergence except in the case of hLDA. † indicates models that do not generate annotated concepts corresponding to WORDNET nodes and hence have no per-node scores. DRR *All* measures the DRR score relative to the entire gold assignment set; *found* measures DRR only for attributes with DRR(w)>0; n is the number of scores averaged.

When prior attribute rank information (Per-Node and CE scores) from the baseline extractions is *not* incorporated, all LDA-based models outperform the unranked baseline (Table 3.1). In particular, LDA yields a 17% reduction in error (MAP) over the baseline, fsLDA yields a 31% reduction, and hLDA yields a 48% reduction (24% reduction over fsLDA). Performance also improves relative to the *ranked* baseline when prior ranking information is incorporated in the LDA-based models, as indicated by CE+Prior scores in Table 3.1. LDA and fsLDA reduce relative error by 6%, and hLDA by 33%. Furthermore, hLDA precision *without* ranking information surpasses the baseline with ranking information, indicating robustness to extraction noise. Overall, learning unconstrained hierarchies (hLDA) increases precision, but as the inferred node distributions do not correspond to WORDNET concepts they cannot be used for annotation.

One benefit to using an admixture model like LDA is that each concept node in the resulting model contains a distribution over attributes specific only to that node (in contrast to, e.g., hierarchical agglomerative clustering). Although absolute precision is lower as more general attributes have higher average precision (Per-Node scores in Table 3.1), these distributions are semantically meaningful in many cases and furthermore can be used to calculate concept assignment precision for each attribute.[9]

[9] Per-node distributions (and hence DRR) were not evaluated for LDA or hLDA, because they are not mapped to WORDNET.

3.5.2 Concept Assignment Precision

The precision of assigning attributes to various concepts is summarized in Table 3.1. Two scores are given: *all* measures DRR relative to the entire gold assignment set, and *found* measures DRR only for attributes with DRR$(w)>0$. Comparing the scores gives an estimate of whether coverage or precision is responsible for differences in scores. fsLDA yields a 20% reduction in relative error (17.2% increase in absolute DRR) over the unranked baseline and a 17.2% reduction (14.2% absolute increase) over the ranked baseline.

3.6 Discussion

This chapter introduced a set of methods based on Latent Dirichlet Allocation (LDA) for filtering attributes and using them to annotate the WordNet ontology. Precision was improved via hierarchical smoothing that takes into account either the local concept structure of WordNet or provides a strong latent tree constraint. LDA significantly outperformed a previous approach both in terms of the concept assignment precision (i.e., determining the correct level of generality for an attribute) and the mean-average precision of attribute lists at each concept (i.e., filtering out noisy attributes from the base extraction set).

CHAPTER 4

Proposed Work

The goal of this proposal is to develop models that can account for the *microstructure* of word relatedness, capturing the multidimensional nature of relations between lexical units and their context-dependence. For example, computing the semantic similarity between *wine* and *vinegar* should only take into account a small number salient features of those concepts, and those features should be quite different from those determining the similarity between *wine* and *bottle*, despite the fact that all three words occur in similar contexts.

Multi-prototype models are an important first step in this direction, however they (1) lack the representational flexibility to adequately model feature saliency, (2) can break down in cases where multiple disparate clusterings of the data are equally feasible and (3) cannot account for multiple dimensions of relatedness. When building lexical semantics models from natural corpora, it is important to address feature noise (cf. §2.6). In high-dimensional clustering problems, a large fraction of features may ultimately be discarded as noise. However, not all features are uniformly noisy: Most features will be relevant for making sense distinctions in certain cases and not in others and hence feature pruning may be too heavy-handed. Xue et al. [121] argue that feature selection should be done per-word in the case of highly polysemous Chinese verbs. The next few sections outline joint clustering and feature selection models that can account for such context-dependent feature relevance, in particular focusing on novel *soft* feature selection methods.

Moving beyond multi-prototype approaches, I propose modeling the full spectrum of relations between words and word-senses using cross-cutting categorization models. Such models can partition the space of word contexts across multiple clusterings, leading to more fine-grained models of contextual dependence, as well as vector-valued word relatedness. Multiple clusterings of a data set (views) are generated using disjoint subsets of the available features (§4.3.3). This model can be extended to jointly partition features *and* data into views, allowing the views to vary in terms of content distribution and salient features (§4.4.1).

Sense proliferation itself may be in part due to the conflation of multiple organizational systems linking the target word to other similar words, leading to a large number of partially overlapping senses. By treating word sense in a multiple clustering framework, these organizational systems can be uncovered and leveraged to build models of per-word semantic generalization. Word senses can be organized along topical, syntactic or operational lines, and different organizational systems account for different subsets of the full set of lexicographical senses. Furthermore, each clustering defines a subset of the available features that are deemed salient, allowing models of lexical semantics the freedom to choose between several relevant subspaces and ignore irrelevant features.

The remainder of this chapter is divided into seven main sections: §4.1 describes several immediate extensions of the multi-prototype model of word meaning, §4.2 develops a set of feature selection and dimensionality reduction models based on LDA, §4.3 develops the suite of novel multi-view models, §4.4 extends the multi-view models to incorporate outlier detection, §4.5 discusses how the similarity metrics for multi-prototype models can be enriched and extended to the multi-view case, §4.6 discusses the merits of applying these models to joint *all-words* models vs. the word type conditional models previously discusses, and §4.7 discusses a large range of potential applications.

4.1 Extending Multi-Prototype Models

The multi-prototype model introduced in chapter 2 comprises the underlying framework for my proposed thesis work. The success of the *combined* approach (§2.4; combining prototypes across multiple clustering scales) indicates that the optimal number of clusters may vary per word. I ropose studying two principled approaches to accounting for automatically assessing clustering capacity:

1. According to Zipf's "Law of Meaning" [123], total word senses are distributed roughly in a power-law, similar to the distribution of word-frequency. Hence, simply allocating representational capacity in the form of additional prototypes proportional to the total number of occurrences may yield optimal meaning representations, trading off expressivity and robustness.

2. Fixing the number of prototypes based on the total number of occurrences can be misleading as the total number of occurrences of a word is heavily corpus-dependent, and in particular semantically "tight" corpora such as WSJ high frequency words may have only a small number of senses actually expressed. Furthermore, the number of clusters should most likely depend on the *variance* of the occurrences, not just the total number.

   A more principled, data-driven approach to selecting the number of prototypes per word is to employ a clustering model with infinite capacity, e.g. the Dirichlet Process Mixture Model [DPMM; 64, 86]. The DPMM assigns positive mass to a variable, but finite number of clusters $\mathbf{z}$,

   $$P(z_i = k|z_{-i}) = \begin{cases} \frac{n_k^{-i}}{\sum_j n_j - 1 + \alpha} & \text{if } n_k > 0 \\ \frac{\alpha}{\sum_j n_j - 1 + \alpha} & k \text{ is a new class.} \end{cases} \tag{4.1}$$

   with probability of assignment to cluster k proportional to the number of data points previously assigned to k, n_k. In this case, the number of clusters no longer needs to be fixed a priori, allowing the model to allocate expressivity dynamically to concepts with richer structure. Such a model would allow naturally more polysemous words to adopt more flexible representations.

   Furthermore, the two-parameter Pitman-Yor generalization of the Dirichlet Process [80] yields power-law distributed cluster sizes,

   $$P(z_i = k|z_{-i}) = \begin{cases} \frac{n_k^{-i} - d}{\sum_j n_j - 1 + \alpha} & \text{if } n_k > 0 \\ \frac{\alpha + dK}{\sum_j n_j - 1 + \alpha} & k \text{ is a new class.} \end{cases} \tag{4.2}$$

with rate proportional to the free parameter d. Since naturally occurring sense *frequences* are also roughly power-law distributed [42], such a model may prove to be a better fit representationally.

4.2 Accounting for Feature Structure

Reisinger and Mooney [88] (chapter 2) demonstrate the importance of feature selection for modeling human word-similarity judgements, finding that the benefits of feature selection often far outweigh the benefits of using more expressive feature representations. However that approach simply discards all but a fixed number of features ordered by weight and does not perform feature selection jointly with clustering; employing more principled feature selection approaches may yield more significant performance gains. This section introduces several such approaches, including two novel unsupervised feature selection procedures based on LDA.

4.2.1 Feature Selection

Moving beyond simple weight-based pruning, employing a feature selection procedure such as latent-factor masking or subspace clustering may yield additional performance gains [38, 43, 47, 75, 95]. Such procedures infer the most likely subset of features given the model and then perform clustering using only those features. The most fundamental approach is simply include a model sparsity term in the clustering optimization criterion. A similar (heuristic) procedure is employed by Pantel and Lin [71], and is shown to be beneficial for word sense discovery.

In addition to simply filtering noisy features, confidence scores generated by the feature-selective clustering model can be used to (1) determine concept- and view-specific discriminative attributes (extending the approach outlined in chapter 3) and (2) provide feedback to upstream information extraction processes, yielding e.g. a measure of precision per extraction pattern (cf. §4.7.4).

Finally, more work should be done exploring the interplay between feature weighting and pruning. Section 2.6 demonstrates how certain feature representations (e.g. χ^2) are more robust to feature noise. However this robustness comes at the cost of being less sensitive to pruning and hence less capable of representing precise lexical semantics. Furthermore, the interplay between feature pruning and dimensionality reduction should be explored more; we expect pruning features *after* dimensionality reduction would have a significantly smaller impact.

4.2.2 Feature Weighting via Topic Models

Moving beyond simple models of feature selection, it is possible to apply the rich structure of LDA-based models to the problem of feature weighting in the context of vector space lexical semantics (cf. Chapter 3). LDA weights features highly when they are commonly found in coherent subsets of the data; such features can be considered sense-specific, and LDA-based feature weighting would thus provide complementary information to, e.g., feature weighting based on the *t-test* or χ^2 criterion.

Although it is capable of identifying useful discriminative features, using the LDA likelihood for feature weighting fails for common, non-discriminative features such as stopwords. When topic interpretability is desired, it is common to first remove stopwords, or to employ an asymmetric Dirichlet prior over topic weights, leading to the formation of a "stopword" topic [116]. However, neither of these approaches addresses the fact that common words often simply have higher likelihood under LDA.

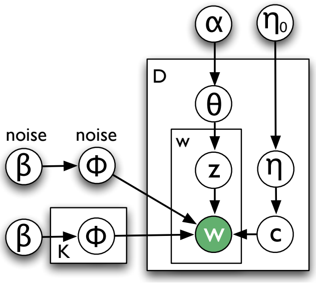
The rest of this section is develops a set of hybrid topic models that perform *soft* feature selection explicitly, resulting in more robust clusterings.

4.2.3 Explicit Feature Selection via Topic Models

I propose a simple feature selective clustering method based on a two-component admixture model, where a document's features are drawn from either a data-dependent mixture model or a single *noise* component. This model is similar structurally to the model proposed by Law et al. [47]. However, instead of allocating entire feature *dimensions* between model and noise components, assignment is done at the level of individual feature occurrences, much like topic assignment in LDA. At a high level, this model can be seen as drawing a document from a combination of a single *prix-fixe* option coupled with (data-independent) *à la carte* choices[1] (Figure 4.1).

Adopting the notation from §3.3.1, the prix-fixe topic model can be written as

$$
\begin{array}{rcllr}
\eta_d|\eta_0 & \sim & \text{Beta}(\eta_0) & d \in D, & \text{(noise prop)} \\
\boldsymbol{\phi}_k|\boldsymbol{\beta} & \sim & \text{Dirichlet}(\boldsymbol{\beta}) & k \in K, & \text{(clusters)} \\
\boldsymbol{\phi}_{\text{noise}}|\boldsymbol{\beta}_{\text{noise}} & \sim & \text{Dirichlet}(\boldsymbol{\beta}_{\text{noise}}) & & \text{(noise)} \\
\boldsymbol{\theta}_d|\boldsymbol{\alpha} & \sim & \text{Dirichlet}(\boldsymbol{\alpha}) & d \in D, & \text{(cluster prop)} \\
c_d|\boldsymbol{\theta}_d & \sim & \text{Mult}(\boldsymbol{\theta}_d) & d \in D, & \text{(cluster ind)} \\
z_{i,d}|\eta_d & \sim & \text{Bernoulli}(\eta_d) & i \in |\mathbf{w}_d|, & \text{(noise ind)} \\
w_{i,d}|\boldsymbol{\phi}_{c_d}, z_{i,d} & \sim & \begin{cases} \text{Mult}(\boldsymbol{\phi}_{\text{noise}}) \\ \quad (z_{i,d} = 1) \\ \text{Mult}(\boldsymbol{\phi}_{c_d}) \\ \quad \text{(otherwise)} \end{cases} & i \in |\mathbf{w}_d|, & \text{(words)}
\end{array}
$$



where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are hyperparameters smoothing the per-document topic distributions and per-cluster word distributions respectively, and η_0 controls the uniformity of the cluster weights.

Each document is drawn from a combination of a single cluster component indicated by c_d and the noise topic. Since the noise topic is shared across all documents, it can account for features with *data-independent* variance, such as stop words and other high-frequency noise. Furthermore, putting an asymmetric prior on $\boldsymbol{\beta}$ yields more fine-grained control over the assumed *uniformity* of the occurrence of noisy features, unlike the model proposed by Law et al. [47]. The likelihood of document d is given by

$$
P(\mathbf{w}_d|\mathbf{z}, c_d, \boldsymbol{\phi}) = \prod_i P(w_{i,d}|\boldsymbol{\phi}_{c_d})^{\delta(z_{d,i}=0)} P(w_{i,d}|\boldsymbol{\phi}_{\text{noise}})^{\delta(z_{d,i}=1)}. \qquad (4.3)
$$

This model can be viewed as a two-topic variant of LDA with the addition of a per-document cluster indicator.[2] By exploiting conjugacy, the latent variables $\boldsymbol{\theta}$, $\boldsymbol{\phi}$ and η can be integrated out, yielding an efficient *collapsed Gibbs sampler*.

[1] Extension to a nonparametric *fancy* Chinese restaurant process is, of course, straightforward.

[2] Specifically, the prix-fixe clustering model is a particular special case of the nested Chinese Restaurant Process with the tree depth fixed to two [10].

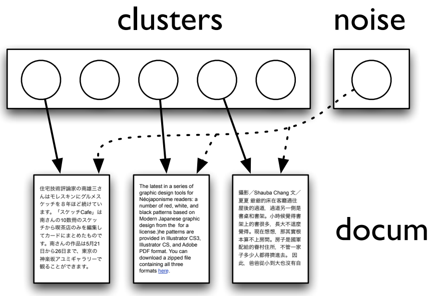


Figure 4.1: Documents are drawn from the *prix-fixe* feature selective clustering model word-by-word, with each word coming either from the document-dependent cluster component or from the document-independent noise component.

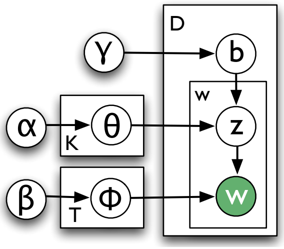
4.2.4 Dense Feature Selection via Bayesian Co-clustering

The textual features employed when clustering word occurrences are high-dimensional and sparse and hence noisy. Feature selection and weighting methods like those proposed in the previous sections address the issue of noise, but do not help combat sparsity, and hence many occurrences can end up with few activated features when using feature selection. However, by performing simultaneous dimensionality reduction and feature selection, both issues can be addressed in a coherent framework. This section outlines a simple Bayesian co-clustering approach for simultaneously reducing feature dimensionality and clustering data and shows how it can be combined with the prix-fixe feature-selective model introduced in the previous section.

Coclustering procedures simultaneously find clusterings of both the rows and the columns of the data matrix, reducing feature dimensionality while grouping data points. Shan and Banerjee [101] introduce a Bayesian coclustering approach based on LDA that allows mixed-membership in both the row and column clustering.

One potential simplification of Shan and Banerjee [101]'s model is to only perform overlap clustering on the features simultaneously with partitioned clustering on the data. The following Bayesian *dense clustering* model clustered documents based on their topic membership proportions:

$$\begin{array}{rcllll}
\gamma|\gamma_0 & \sim & \text{Dirichlet}(\gamma_0), & & \text{(cluster proportio} \\
\boldsymbol{\theta}_k|\boldsymbol{\alpha} & \sim & \text{Dirichlet}(\boldsymbol{\alpha}), & k \in K, & \text{(topic proportions} \\
\boldsymbol{\phi}_t|\boldsymbol{\beta} & \sim & \text{Dirichlet}(\boldsymbol{\beta}), & t \in T, & \text{(topics)} \\
b_d|\boldsymbol{\gamma} & \sim & \text{Mult}(\boldsymbol{\gamma}), & d \in D, & \text{(cluster indicator)} \\
z_{i,d}|\boldsymbol{\theta}_d, b_d & \sim & \text{Mult}(\boldsymbol{\theta}_{b_d}), & i \in |\mathbf{w}_d|, & \text{(topic indicator)} \\
w_{i,d}|\boldsymbol{\phi}_{z_{i,d}} & \sim & \text{Mult}(\boldsymbol{\phi}_{z_{i,d}}), & i \in |\mathbf{w}_d|, & \text{(words)}
\end{array}$$



In this model K groups of documents share the same topic proportions $\boldsymbol{\phi}_k$ (i.e. cluster centroids), corresponding to hard-clustering. This model reduces to LDA when $K \to D$, i.e. each document is assigned to its own cluster, and hence is more computationally efficient than LDA, despite performing clustering and topic-modeling jointly.

Combining the dense clustering model with the prix-fixe feature-selective model proposed in the previous section yields a coherent framework for joint dimensionality reduction, feature selection and clustering, i.e. *dense* feature-selective clustering.

4.3 Feature Partitioning via Cross-cutting Categorization

In addition to feature selection, I propose to study the effects of three *multiple-clustering* models based on cross-cutting categorization (cross-cat), which find several clusterings of the data (views) each using different subsets of features. Cross-cat is an effective way to control the effects of features unrelated to any one particular clustering scheme [56, 57] and hence solves one of the basic problems with exemplar and multi-prototype models using raw textual features.

Different subsets of features may yield different sense views; e.g. clustering using only syntactic features vs. clustering using only document co-occurrence features. Psychologically, humans use overlapping taxonomies to organize conceptual information in many domains; i.e. foods can be organized situationally, *breakfast food*, *dinner food*, *snack*, etc, or by their type, *dairy*, *meat*, etc. Each organization system may have different salient features. Cross-cat models account for this structure by assigning concept features to one of several views, clustering the data separately with each view. This approach yields multiple orthogonal clusterings and isolates the effects of noisy features.

As feature dimensionality increases, the number of ways the data can exhibit interesting structure goes up exponentially. Multiple clustering based on cross-cat is one approach to inferring feature subspaces that lead to high quality data partitions. The cross-cat models differs significantly from, e.g., the multiple disparate clusterings framework proposed by Jain et al. [40]. In that work, all clusterings use all features, and hence robustness to feature noise is not treated. Cross-cat is more similar to the model proposed by Cui et al. [17], which generates a maximally orthogonal cluster ensemble [cf. 5, 104]. The data are repeatedly projected onto the space most orthogonal to the current clustering and then reclustered.

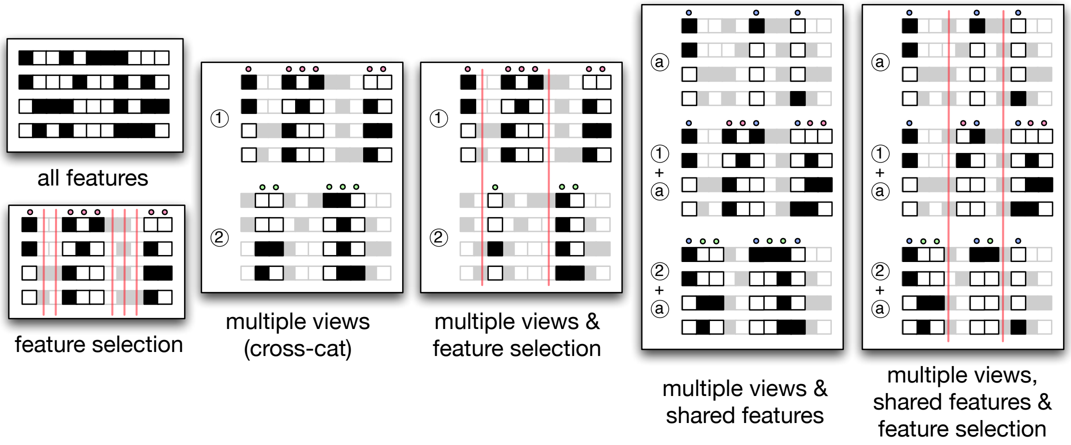
I propose a suite of unsupervised methods for determining feature relevance, extending model-based feature-selection and cross-cat to account for feature-sharing between multiple competing categorization models. These models are aimed at overcoming the main limitation of cross-cat, allowing informative features to be shared by several views. The first extension, multiple-views with *shared features*, allows each view to inherit a set of shared features in addition to its view-specific features. The second extension, *factorial feature allocation* (FFA), puts the entire binary feature assignment matrix $\mathbf{Z}$ under the control of the model, treating it as a random variable. Figures 4.2, 4.3, and 4.4 summarize the various model combinations considered.

I first establish some notation and present the original cross-cat model.

- $d \in [1 \dots D]$ F-dimensional data vectors $\mathbf{w} = [\mathbf{w}_1, \dots, \mathbf{w}_D]^\top$.
- $m \in [1 \dots M]$ views defined by $\mathbf{Z}$. View m is a binary vector specifying which features are included in the mth clustering.
- $k \in [1 \dots K_m]$ clusters in clustering $m \in [1 \dots M]$, $\mathbf{c}_k^m$.

Define the unary *factorial feature projection operator*

$$(\star\mathbf{Z}_{\cdot,m}) : \mathbb{R}^F \to \mathbb{R}^{||\mathbf{Z}_{\cdot,m}||_1}, \tag{4.4}$$



all features

feature selection

multiple views (cross-cat)

multiple views & feature selection

multiple views & shared features

multiple views, shared features & feature selection

Figure 4.2: Progression of proposed feature selection and multi-view models. Horizontal vectors indicate data; circled numbers and letters represent disparate views; grayed boxes indicate features not present in that particular view; and vertical lines represent features removed from all views. Clustering occurs separately within each view. In the case of shared feature views, features assigned to view (a) are present in all views.

mapping data vectors of dimension F to vectors with dimension equal to the number of nonzero entries of the column-vector $\mathbf{Z}_{\cdot,m}$ (i.e. $||\mathbf{Z}_{\cdot,m}||_1$). Let

$$\lambda^m \overset{\text{def}}{=} \{j : j \in [1 \dots F], [\mathbf{Z}]_{j,m} = 1\} \tag{4.5}$$

be the ordered indices of the nonzero entries of $\mathbf{Z}_{\cdot,m}$ and let $L^m \overset{\text{def}}{=} |\lambda^m| = ||\mathbf{Z}_{\cdot,m}||_1$ be the number of nonzero entries. Then define

$$\mathbf{w} \star \mathbf{Z}_{\cdot,m} \overset{\text{def}}{=} (w_{\lambda_1^m}, \dots, w_{\lambda_{L^m}^m})^\top, \tag{4.6}$$

i.e. the projection of $\mathbf{w}$ onto the lower-dimensional subspace specified by the nonzero entries of $\mathbf{Z}_{\cdot,m}$. Finally $\mathbf{w}^{\star(m)}$ will be used as shorthand for $\mathbf{w} \star \mathbf{Z}_{\cdot,m}$ when the view assignment matrix $\mathbf{Z}$ is unambiguous.

All models discussed in this section can be written in the form

$$\begin{aligned} P(\mathbf{Z}, \mathbf{c}|\mathbf{w}) &\propto P(\mathbf{Z}, \{\mathbf{c}^m\}, \mathbf{w}) && (4.7) \\ &= P(\mathbf{Z}) \prod_{m=1}^{M} P(\mathbf{w}^{\star(m)}|\mathbf{c}^m) P(\mathbf{c}^m). && (4.8) \end{aligned}$$

where $P(\mathbf{Z})$ is the prior distribution on views and $P(\mathbf{c}^m)$ is a prior on the clustering for view m, e.g. the DPMM, and $P(\mathbf{w}^{\star(m)}|\mathbf{c}^m)$ is the likelihood of the data $\mathbf{w}$ restricted to the feature subset $\mathbf{Z}_{\cdot,m}$ given the corresponding clustering $\mathbf{c}^m$ [100].

4.3.1 Cross-Cat

In the standard cross-cat model, $P(\mathbf{Z})$ is constructed by first drawing the vector $\tilde{\mathbf{z}} \sim \text{CRP}(\alpha)$, i.e. assigning each feature to some view via the *Chinese Restaurant Process* [79].

$\mathbf{Z}$ is then derived from $\tilde{\mathbf{z}}$ in the obvious way: each feature (row vector of $\mathbf{Z}$) has only a single nonzero entry corresponding to the column index of the view it is assigned to via $\tilde{\mathbf{z}}$:

$$[\mathbf{Z}]_{f,m} = \begin{cases} 1 & \tilde{\mathbf{z}}_f = m, \\ 0 & \text{otherwise.} \end{cases} \tag{4.9}$$

The cross-cat model is capable of finding disjoint views with maximally probable clusterings. The Dirichlet Process parameter α on view assignment controls the trade-off between the fit of any one clustering and the cost of adding an addition clustering, taking features away from the others. Because views form hard partitions of features, cross-cat is not capable of representing *all* of the most probable clusterings simultaneously, i.e. features cannot be shared across views. I address this limitations in the new proposed models.

4.3.2 Shared Feature Partitions

The first novel extension of cross-cat adds an additional *shared* view that specifies features conserved across all views (Figure 4.2). This puts pressure on the model to identify the most information / most generic features to conserve across clusterings. The shared features themselves do not constitute a separate clustering, and hence do not necessarily need to yield good views on their own. The remaining view-specific features capture the individual idiosyncrasies of each clustering.

The shared feature model is capable of identifying features that contribute to multiple clusterings of the data and hence may find exactly the features that characterize the strongest sense distinctions. For example, features that contribute both to syntactic sense clustering and topical sense clustering. Thus the shared feature model can be viewed a form of *robust clustering*, finding the commonalities between an ensemble of orthogonal clusterings.

The shared view is encoded using an additional random binary vector $\mathbf{u}$ with one entry per feature, indicating whether that feature should be included in all views or not. The resulting construction for $\mathbf{Z}$ is then

$$[\mathbf{Z}]_{f,m} = \begin{cases} 1 & \tilde{\mathbf{z}}_f = m \\ u_f & \text{otherwise.} \end{cases} \tag{4.10}$$

where, again $\tilde{\mathbf{z}} \sim \text{CRP}(\alpha)$, and e.g.,

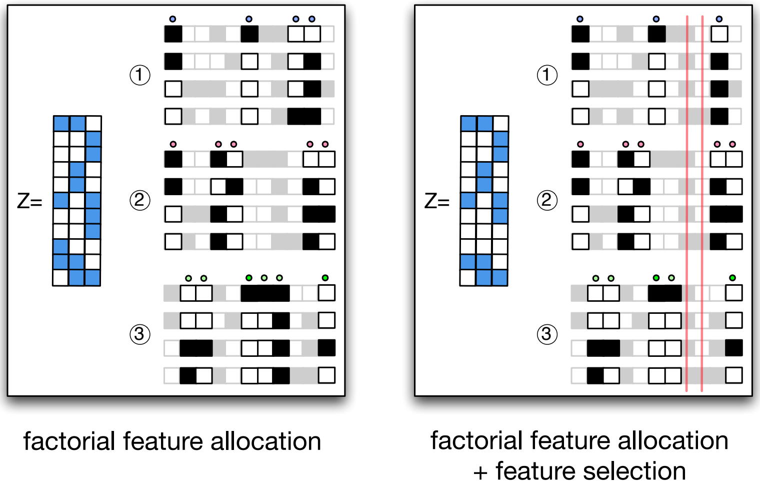
$$u_f|\mu_f \sim \text{Bernoulli}(\mu_f) \tag{4.11}$$

$$\mu_f|\xi \sim \text{Beta}(\xi). \tag{4.12}$$

A similar result could be realized by reserving one cluster in $\tilde{\mathbf{z}}$ to indicate whether the feature is shared or not, however the likelihood structure of this model may cause the sampler not to mix well. However, it may be possible to implement this model using the *colored stick-breaking process* which allows for both exchangeable and non-exchangeable partitions, improving efficiency [29].

From a data-analytic perspective this model is interesting because the shared features may capture some intuitive basic structure specific to the particular word, e.g., some notion of the underlying metaphor structure of *line* independent of topical variation.

Klein and Murphy [44] find no psychological evidence for shared structure linking different senses of polysemous words, indicating that the shared structure model may not



factorial feature allocation

factorial feature allocation + feature selection

Figure 4.3: Factorial feature allocation model. Horizontal vectors indicate data; circled numbers represent disparate views; grayed boxes indicate features not present in that particular view; and vertical lines represent features removed from all views. The $F \times K$ dimensional matrix $\mathbf{Z}$ specifies what features are present in what view.

perform well relative to the other models proposed here. However, their experiments were not focused on fine-grained sense distinctions such as those present in WordNet, and furthermore this does not necessarily indicate that such models are not applicable to lexical semantics: when deriving occurrence features from raw text, it is expected that there is some feature overlap attributable to the "background" meaning of the word.

4.3.3 Factorial Feature Allocation

Factorial feature allocation puts the full feature-to-view map $\mathbf{Z}$ under the control of the model (Figure 4.3). With FFA each feature is assigned to some subset of the available views, with some probability. The *Indian Buffet Process* provides a suitable nonparametric prior for FFA, where draws are random binary matrices with a fixed number of rows (features) and possibly an infinite number of columns [30]. Note that in our case the "latent" feature dimensions inferred by the IBP correspond to feature views in the original clustering problem.

$$\mathbf{Z}|\theta_{\mathbf{Z}} \sim \text{IBP}(\theta_{\mathbf{Z}}) \tag{4.13}$$

which yields feature-to-view assignments where each feature occurs in $A_f \sim \text{Poisson}(\theta)$ views ($\mathbb{E}[A_f] = \theta$), and the total number of views $M \sim \text{Poisson}(\theta H_{|\mathbf{w}|})$[3]

The main benefit of factorial feature allocation over the simpler models is that features can be shared arbitrarily between views, with the IBP specifying only a prior on the *number* of features active in any one view. Concretely, this allows the model to simultaneously represent the most probable clusterings using $\mathbb{E}[\theta F]$ features. Since M is low for most applications considered, factorial feature allocation is not significantly more complex computationally than the shared feature model.

[3] $H_n \overset{\text{def}}{=} \sum_{i=1}^{n} \frac{1}{i}$ is the nth harmonic number.

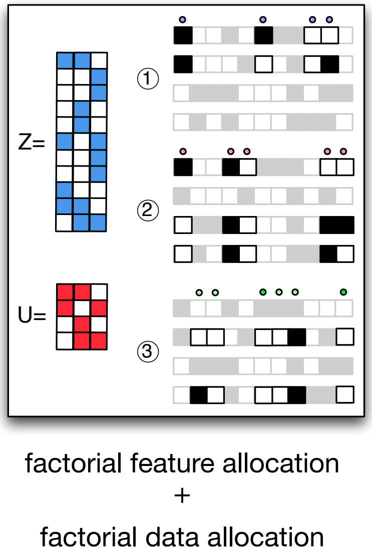


Figure 4.4: Factorial feature and data allocation model. The $F \times K$ dimensional matrix $\mathbf{Z}$ specifies what features are present in what view; the $D \times K$ dimensional matrix $\mathbf{U}$ specifies what data points are allocated to each view.

Finally, I propose to explore to what extent to which topic models are similar to factorial feature allocation; at a high level, factorial feature allocation can be viewed as a type of topic model where each topic has only a single word/feature, and may be related to the class of *Focused Topic Models* [118]. Exploring this duality should lead to more efficient sampling methods for FFA, as well as topic models better able to capture latent feature structures. Also, it would allow the development of FFA models with latent hierarchical structure, based on e.g. the nested Chinese Restaurant Process [10], labeled LDA [83] or the Kingman's coalescent [107].

4.4 Accounting for Data Structure

The dual problem to feature selection is determining data relevance, i.e. removing outliers or irrelevant data points. Previous lexical semantic models such as *Clustering by Committee* use ad-hoc criteria for improving robustness to outliers [74]; in Statistics, outliers are treated in density estimation using robust distributions, e.g. Laplace [16]. In addition to studying the applicability of *background cluster* models such as the colored stick-breaking process to clustering word occurrences [29], I propose extending the FFA model described in §4.3 to jointly model feature and data allocation (FFDA; Figure 4.4)).

4.4.1 Joint Factorial Feature and Data Allocation

FFDA allocates features and data jointly among disparate views, leveraging the assumption that subsets of the data are better fit by subsets of the available features. From the standpoint of concept organization, this corresponds to different organizational schemes acting on different subsets of the available concepts (i.e. not all concepts are shared across all organizational schemes). For example, when organizing animals by their scientific properties (e.g., *habitat, taxonomy, gestation period*) it makes sense to exclude fictional counterparts (e.g., fictional ducks such as *Donald*); however, when organizing them by their apparent physical properties (flies, quacks, has feathers), perhaps fictional animals should be included.

In the context of lexical semantics, FFDA can be motivated by considering *differential feature noise*: i.e. assumption that some features are content bearing for some subsets of data, but not for others. Identifying *when* a particular feature is spurious requires considering it in the context of the other features, and this is not a strongpoint of traditional clustering analysis.

FFDA can be defined by simply augmenting FFA with an additional random binary matrix $\mathbf{U}$ specifying which data points are included in which views:

$$\begin{aligned} \mathbf{Z}|\theta_{\mathbf{Z}} &\sim \text{IBP}(\theta_{\mathbf{Z}}) && (4.14) \\ \mathbf{U}|\theta_{\mathbf{U}} &\sim \text{IBP}(\theta_{\mathbf{U}}) && (4.15) \end{aligned}$$

where $\mathbf{U}$ has dimension $D\times M$. Ensuring that $\mathbf{Z}$ and $\mathbf{U}$ have the same column dimensionality (number of views) can be achieved by drawing a larger matrix of dimension $(D+F)\times M$ from the IBP and partitioning it into $\mathbf{Z}$ and $\mathbf{U}$. Note that joint data and feature allocation does not significantly raise the computational complexity of the model above that of factorial feature allocation.

FFDA also requires redefining the projection operator in Equation 4.4 to operate over both the rows and columns of the data matrix, which can be realized in the obvious way. The joint operator will be written as $\mathbf{w} \circledast (\mathbf{Z}_{\cdot,m}, \mathbf{U}_{\cdot,m})$ with the shorthand $\mathbf{w}^{\circledast(m)}$ when the feature allocation and data allocation matrices are unambiguous.

Finally, the form of the general probabilistic model must be extended to include $\mathbf{U}$ and the dependence of $\mathbf{c}$ on the data partition:

$$\begin{aligned} P(\mathbf{Z}, \mathbf{U}, \mathbf{c}|\mathbf{w}) &\propto P(\mathbf{Z}, \mathbf{U}, \{\mathbf{c}^m\}, \mathbf{w}) && (4.16) \\ &= P(\mathbf{Z})P(\mathbf{U}) \prod_{m=1}^{M} P(\mathbf{w}^{\circledast(m)}|\mathbf{c}^{\circledast(m)})P(\mathbf{c}^{\circledast(m)}). && (4.17) \end{aligned}$$

There are several ways to account for the fact that $\mathbf{c}$ depends on $\mathbf{U}$. The simplest to extend the prior $P(\mathbf{c})$ to the entire data set, but to restrict the likelihood $P(\mathbf{w}^{\circledast(m)}|\mathbf{c}^{\circledast(m)})$ to only the data contained in the view.

4.5 Multiple Cluster Similarity Metrics

AvgSim and MaxSim (§2.2.2) are conceptually simple methods for computing similarity between mixture models, however they do not necessarily respect the structure of the underlying probabilistic models, and furthermore present nontrivial difficulties when applied to multi-view models. I propose to revisit the space of potential similarity metrics defined over mixture models in order to develop a coherent, tractable notion of similarity for the structured models introduced above. In particular, building distance metrics based on the family of f-divergences

$$D_f(P||Q) = \int_{\Omega} f\left(\frac{dP}{dQ}\right) dQ$$

e.g. discrete KL-divergence [52]

$$D_{\mathrm{KL}}(P||Q) = \sum_{x \in \Omega} P(x) \log \frac{P(x)}{Q(x)}$$

or methods based on probability measures such as Bhattacharyya distance

$$D_{\mathrm{B}}(p, q) = -\log \left(\sum_{x \in \Omega} \sqrt{p(x)q(x)} \right)$$

both of which are related to the class of Renyi divergences [90]. However, these are general measure of statistical similarity and do not take into account the structure of the proposed models. For example, in the multi-view model, one potential similarity metric first determines the most similar views for each word and then computes the mixture distance between the selected views (i.e. combining both AvgSim and MaxSim). Alternatively, multi-view clusterings can be collapsed into a single view using standard cluster ensemble methods [104].

4.6 Word-Joint Models

All of the previously introduced models can be fit to word/phrase occurrence or concept data either (1) conditional on the word-type, uncovering usage patterns for a single word or (2) joint across all words, clustering words with similar usages together. The former model is more computationally tractable and can be parallelized naively by word-type. However, since this method independently clusters the contexts of each word, the usages discovered for w cannot influence the usages discovered for $w' \neq w$. Sharing statistical strength across similar words could yield better results for rarer words, in addition to providing a more coherent model of human conceptual organization. Furthermore, the word-joint model automatically computes inter-word similarity, obviating the need for defining similarity metrics on multiple clusterings.

Another benefit of the cross-cutting model is that it can de-aggregate context vectors, accounting for polysemy even when multiple senses have been encoded in the same feature vector. For example, when clustering *apple* with other fruits, cross-cat might find certain features such as *stock* or *company* to be irrelevant, ignoring the homonymous usage.

4.7 Applications

I propose evaluating cross-cutting models in several downstream applications, including lexical semantics tasks such as selectional preference and paraphrase identification (§4.7.1), modeling the overlapping hierarchical structure of "folksonomies" such as Wikipedia (§4.7.2), associative anaphora resolution (§4.7.3), knowledge acquisition (§4.7.4), and text classification (§4.7.5).

4.7.1 Lexical Semantics

Selectional Preference

The selectional preference of verbs has been studied extensively in the context of distributional lexical semantics [9, 69, 91], including extensions to multi-prototype representations of arguments [73]. Ritter et al. [92] demonstrated significant gains from applying LDA to jointly model the selectional preference of a large subset of TextRunner relations. Moving beyond LDA, cross-cutting categorization models could potentially improve models of selectional preference by identifying feature subsets describing the relations governing the arguments of target word.

Paraphrase Identification

Previous approaches to paraphrase identification (e.g. [61]) can fail because they do not take into account context-dependence, and often paraphrases are only accurate for a subset of meanings of the original phrase [59]. Thus, the multi-prototype model could potentially lead to more robust paraphrase identification (cf. [23]). In particular, identifying when it is possible to use a paraphrase, i.e. the particular set of valid contexts its inherently a clustering problem, and only a small percentage of feature dimensions are actually relevant to the task, suggesting that feature partitioning would be a powerful approach.

4.7.2 Hierarchical Cross-Categorization

Understanding the internal feature representations of concepts and how it comes to bear on conceptual organization and pragmatics is important for computational linguistic tasks that require a high degree of semantic knowledge: e.g. information retrieval, machine translation, and unsupervised semantic parsing. Since reported properties are cognitively salient and discriminative, extracting them would be helpful for semantic search tasks such as query disambiguation and user intent modeling. Furthermore, feature norms have been used to understand the conceptual information people possess for the thematic roles of verbs [24].

Combining hierarchical topic decompositions introduced in chapter 3 with cross-cat yields a coherent framework for mixtures of overlapping ontologies. Current fixed ontology models of conceptual organization such as WordNet cannot easily capture such phenomena [89], although there is significant evidence for multiple organizational principles in Wikipedia categories [87]; for example people are organized by their occupation (e.g. *American politicians*), their location (e.g. *People from Queens*), or chronology (e.g. *1943 births*). Likewise, most ducks can *fly* and *quack* but only fictional ducks *appear in cartoon*s or *have nephews*; does this mean fictional ducks can be *blanched in water* and *air dried*? Accounting for the structure of such natural “tangled hierarchies,” or “folksonomies,” requires significantly richer models.

I propose extending the cross-categorization model to *latent* hierarchical data, which requires defining a consistent model of multiple overlapping *local* categorizations within a larger hierarchical structure. Preliminary work on this model suggests that it better separates attributes according to their usage domains. Practical applications include noise-filtering for open-domain category and attribute extraction (Chapter 3), as well as determining what terms/features are most relevant to certain query modes (classifying query intent). Evaluation of the underlying prediction models can be carried out using human annotators recruited from Mechanical Turk.

Hierarchical cross-categorization would also benefit significantly from data partitioning, as one would not expect every feature view to be relevant to *all* concepts in Wikipedia. Instead, organizational frames have a native level of generality over which they operate, controlling what concepts are relevant to include.

4.7.3 Associative Anaphora Resolution

Associative anaphora[4]: are a type of bridging anaphora with the property that the anaphor and its antecedent are not coreferent, e.g.,

1. Once she saw that all **the tables**$_{(\rightsquigarrow 1)}$ were taken and **the bar**$_{(\rightsquigarrow 1)}$ was crowded, she left **the restaurant**$_{(1)}$.
2. Shares of **AAPL**$_{(2)}$ closed at $241.19. **Volatility**$_{(\leftsquigarrow 2)}$ was below the 10-day moving average.

where *tables* and *bar* in example as aspects of the *restaurant* and *volatility* in example 2 is an aspect of *AAPL* [15]. Resolving associative anaphora naturally requires access to richer semantic knowledge than resolving e.g. indirect anaphora, where the anaphor and its antecedent differ only by reference and can be resolved syntactically [13, 97]. The smoothed property extraction methods presented in chapter 3 could provide a basis for performing associative anaphora resolution, hence I propose an evaluation combining it with existing coreference resolution systems [e.g. 34].

Resolving associative anaphora is another domain that might potentially benefit from multi-language models. The fundamental semantic (mereological) relationships are conserved across languages, and hence resource-rich languages could be adapted for use in resource-poor languages. Note how this contrasts sharply with purely syntax-level tasks, such as coreference resolution, where knowledge of the particular language structure is necessary.

4.7.4 Knowledge Acquisition

Vector-space models are commonly used in knowledge acquisition (KA), e.g. for attribute and class-instance acquisition [51, 72, 111], and hence could benefit from multi-prototype and multi-view extensions, identifying relevant axes of variation along which additional high-quality data can be extracted. The current state of the art in KA ignores the downstream uses of its data, likewise, machine learning (ML) models are typically unaware of the details of the upstream KA system that generated the data. Although such functional modularity greatly simplifies system-level development, a significant amount of information is discarded that could greatly improve both systems. Several general-purpose frameworks for integrating KA and ML have been recently proposed, relying on particular model-[58] or structural assumptions [14]. For this project, I propose a much simpler approach: leveraging generative models of the data to predict the likelihood of specific instances or features being outliers. Such approaches are common in the statistics literature [38, 113] but find little traction in KA.

4.7.5 Text Classification and Prediction

One straightforward way to evaluate lexical semantics models is to embed features derived from them in existing text classification and prediction problems. Comparing results to existing baselines gives a rough measure of how much additional useful semantic content is captured for that domain. Towards this end I propose evaluating the lexical semantics models on sentiment analysis [70] and predicting properties of financial text [46].

[4] Also referred to as *mereological anaphora*, cf. Poesio et al. [81].

4.7.6 Speculative Work

Cross-Lingual Property Generation

Chapter 2 introduced salient property prediction as a specific application of structured lexical semantic models. Such properties are useful in downstream applications such as associative anaphora resolution (§4.7.3), but can also be evaluated on their own, e.g. comparing against human property generation norms [60]. Extending these models with multiple prototypes and factorial feature association is a logical next step, and would provide a coherent framework for addressing cross-language differences in concept organization.

Modeling concept structure across multiple languages simultaneously would help mitigate the noise introduced by per-language extraction idiosyncrasies and leveraging resource-rich languages to improve inference for resource-poor languages. Furthermore a large-scale comparison of concept organization norms across languages would shed light on important aspects of cross-cultural pragmatics [117].

Twitter

Twitter is a rich testbed for identifying and understanding the root causes of modern language evolution: Denotative shifts in meaning can be correlated with current events and tracked in real time. Furthermore, standardized internet-specific language features such as topical hash-tags are developing at a rapid pace, incubated primarily on Internet blogs and Twitter.

Due to its high degree of fluidity in term usage and unusually short context lengths [78, 85], traditional lexical semantics models may fail to capture interesting phenomena on Twitter. I propose applying the robust, structured models developed in this thesis to modeling the real-time lexical semantic development of Twitter hashtags. In particular, models based on DPMMs can adapt to form new clusters in real time when new data is added that does not fit well with the existing inferred structure. This ability is important since it is impossible to fix the capacity of lexical semantic models *a priori*, as new concepts (denoting current events) are constantly being added to the lexicon.

CHAPTER 5

Conclusion

This proposal has outlined the application of *cross-cutting* categorization models to distributional lexical semantics, focusing on its ability to (1) account for feature noise and (2) extract coherent feature subsets that define similarity relations between words. Cross-cutting models are able to succinctly account for the notion that humans rely on different categorization systems for making different kinds of generalizations. These latent categorization systems underly lexical semantic phenomenon such as contextual and selectional preference, and hence modeling them may yield significant improvements in machine translation and information retrieval. Furthermore, cross-cutting models can be naturally extended to model hierarchical data, inferring multiple overlapping ontologies. Such structures can be leveraged to improve, e.g., open-domain attribute and relation extraction.

Bibliography

[1] E. Agirre and P. Edmonds. *Word Sense Disambiguation: Algorithms and Applications (Text, Speech and Language Technology)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[2] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Paşca, and A. Soroa. A study on similarity and relatedness using distributional and Wordnet-based approaches. In *Proc. of NAACL-HLT-09*, pages 19–27, 2009.

[3] J. Anderson. *The adaptive character of thought*. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1990.

[4] F. G. Ashby and L. A. Alfonso-Reese. Categorization as probability density estimation. *J. Math. Psychol.*, 39(2):216–233, 1995.

[5] J. Azimi and X. Fern. Adaptive cluster ensemble selection. In *IJCAI'09: Proceedings of the 21st international jont conference on Artifical intelligence*, pages 992–997, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[6] L. D. Baker and A. K. McCallum. Distributional clustering of words for text classification. In *Proceedings of 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 96–103, Melbourne, Australia, 1998.

[7] A. Banerjee, I. Dhillon, J. Ghosh, and S. Sra. Clustering on the unit hypersphere using von Mises-Fisher distributions. *Journal of Machine Learning Research*, 6: 1345–1382, 2005.

[8] M. Baroni, B. Murphy, E. Barbu, and M. Poesio. Strudel: A corpus-based semantic model based on properties and types. *Cognitive Science*, 34, 2010.

[9] S. Bergsma, D. Lin, and R. Goebel. Discriminative learning of selectional preference from unlabeled text. In *EMNLP '08: Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 59–68, Morristown, NJ, USA, 2008. Association for Computational Linguistics.

[10] D. Blei, T. Griffiths, M. Jordan, and J. Tenenbaum. Hierarchical topic models and the nested Chinese restaurant process. In *Proceedings of the 17th Conference on Neural Information Processing Systems (NIPS-2003)*, pages 17–24, Vancouver, British Columbia, 2003.

[11] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:2003, 2003.

[12] A. Budanitsky and G. Hirst. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47, 2006.

[13] R. Bunescu. Associative anaphora resolution: A web-based approach. In *In Proceedings of the EACL2003 Workshop on the Computational Treatment of Anaphora*, pages 47–52, 2003.

[14] R. Bunescu. Learning with probabilistic features for improved pipeline models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-08)*, 2008.

[15] M. Charolles. Associative anaphora and its interpretation. In *Journal of pragmatics*, volume 31, 1999.

[16] A. Cord, C. Ambroise, and J.-P. Cocquerez. Feature selection in robust clustering based on laplace mixture. *Pattern Recogn. Lett.*, 27(6):627–635, 2006.

[17] Y. Cui, X. Z. Fern, and J. G. Dy. Non-redundant multi-view clustering via orthogonalization. In *ICDM '07: Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*, pages 133–142, Washington, DC, USA, 2007. IEEE Computer Society.

[18] J. Curran. *From Distributional to Semantic Similarity*. PhD thesis, University of Edinburgh, 2004.

[19] J. R. Curran. *From Distributional to Semantic Similarity*. PhD thesis, University of Edinburgh. College of Science, 2004.

[20] J. R. Curran and M. Moens. Improvements in automatic thesaurus extraction. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition*, pages 59–66, Morristown, NJ, USA, 2002. Association for Computational Linguistics.

[21] I. S. Dhillon and D. S. Modha. Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42:143–175, 2001.

[22] K. Erk. A simple, similarity-based model for selectional preferences. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*. Association for Computer Linguistics, 2007.

[23] K. Erk and S. Pado. Exemplar-based models for word meaning in context. In *Proceedings of ACL*, 2010.

[24] T. Ferretti. Integrating verbs, situation schemas, and thematic role concepts. *Journal of Memory and Language*, 44(4):516–547, May 2001.

[25] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. Placing search in context: the concept revisited. In *Proc. of the 10th international conference on World Wide Web*, 2001.

[26] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proc. of IJCAI-07*, pages 1606–1611, 2007.

[27] W. Gao, C. Niu, J. Nie, M. Zhou, J. Hu, K. Wong, and H. Hon. Cross-lingual query suggestion using query logs of different languages. In *Proceedings of the 30th ACM Conference on Research and Development in Information Retrieval (SIGIR-07)*, pages 463–470, Amsterdam, The Netherlands, 2007.

[28] D. Graff. *English Gigaword*. Linguistic Data Consortium, Philadephia, 2003.

[29] P. J. Green. Colouring and breaking sticks: Random distributions and heterogeneous clustering. In *arXiv:1003.3988*, 2010.

[30] T. Griffiths and Z. Ghahramani. Infinite latent feature models and the Indian buffet process. In *Advances in Neural Information Processing Systems 18*, pages 475–482. MIT Press, Cambridge, MA, 2006.

[31] T. Griffiths and M. Steyvers. A probabilistic approach to semantic representation. In *Proceedings of the 24th Conference of the Cognitive Science Society (CogSci02)*, pages 381–386, Fairfax, Virginia, 2002.

[32] T. Griffiths, T. L. Griffiths, M. Steyvers, and J. B. Tenenbaum. Topics in semantic representation. *Psychological Review*, 114:2007, 2007.

[33] T. L. Griffiths, K. R. Canini, A. N. Sanborn, and D. J. Navarro. Unifying rational models of categorization via the hierarchical Dirichlet process. In *Proc. of CogSci-07*, 2007.

[34] A. Haghighi and D. Klein. Unsupervised coreference resolution in a nonparametric Bayesian model. In *Proc. ACL 2007*, pages 848–855. Association for Computational Linguistics, 2007.

[35] Z. Harris. Distributional structure. *Word*, 10(23):146–162, 1954.

[36] M. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, pages 539–545, Nantes, France, 1992.

[37] E. Heit and J. Rubinstein. Similarity and property effects in inductive reasoning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 20(2):411–422, 1994.

[38] P. D. Hoff. Model-based subspace clustering. *Bayesian Analysis*, 1(2):321–344, 2006.

[39] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd ACM Conference on Research and Development in Information Retrieval (SIGIR-99)*, pages 50–57, Berkeley, California, 1999.

[40] P. Jain, R. Meka, and I. S. Dhillon. Simultaneous unsupervised learning of disparate clusterings. In *SDM*, pages 858–869. SIAM, 2008.

[41] B. J. Jansen, D. L. Booth, and A. Spink. Determining the user intent of web search engine queries. In *Proc. of WWW 2007*. ACM, 2007.

[42] A. Kilgarriff. How dominant is the commonest sense of a word. In *In Proceedings of Text, Speech, Dialogue*, pages 1–9. Springer-Verlag, 2004.

[43] S. Kim, M. G. Tadesse, and M. Vannucci. Variable selection in clustering via Dirichlet process mixture models. *Biometrika*, 93(4):877–893, 2006.

[44] D. E. Klein and G. L. Murphy. The representation of polysemous words. *Journal of Memory and Language*, 45:259–282, 2001.

[45] D. E. Klein and G. L. Murphy. Paper has been my ruin: Conceptual relations of polysemous senses. *Journal of Memory and Language*, 47:548570, 2002.

[46] S. Kogan, D. Levin, B. R. Routledge, J. S. Sagi, and N. A. Smith. Predicting risk from financial reports with regression. In *NAACL '09: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 272–280, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[47] M. H. C. Law, A. K. Jain, and M. A. T. Figueiredo. Feature selection in mixture-based clustering. In *Advances in Neural Information Processing Systems 15*, pages 625–632, 2002.

[48] L. Lee. Measures of distributional similarity. In *37th Annual Meeting of the Association for Computational Linguistics*, pages 25–32, 1999.

[49] W. Li and A. McCallum. Pachinko allocation: DAG-structured mixture models of topic correlations. In *Proceedings of the 23rd International Conference on Machine Learning (ICML-06)*, pages 577–584, Pittsburgh, Pennsylvania, 2006.

[50] D. Lin and P. Pantel. Concept discovery from text. In *Proceedings of the 19th International Conference on Computational linguistics (COLING-02)*, pages 1–7, Taipei, Taiwan, 2002.

[51] D. Lin, S. Zhao, L. Qin, and M. Zhou. Identifying synonyms among distributionally similar words. In *Proceedings of the Interational Joint Conference on Artificial Intelligence*, pages 1492–1493. Morgan Kaufmann, 2003.

[52] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37:145–151, 1991.

[53] B. C. Love, D. L. Medin, and T. M. Gureckis. SUSTAIN: A network model of category learning. *Psych. Review*, 111(2):309–332, 2004.

[54] W. Lowe. Towards a theory of semantic space. In *Proceedings of the 23rd Annual Meeting of the Cognitive Science Society*, pages 576–581, 2001.

[55] X. Ma, J. Boyd-Graber, S. S. Nikolova, and P. Cook. Speaking through pictures: Images vs. icons. In *ACM Conference on Computers and Accessibility*, 2009.

[56] V. K. Mansinghka, C. Kemp, and J. B. Tenenbaum. Structured priors for structure learning. In *Proc. UAI 2006*. AUAI Press, 2006.

[57] V. K. Mansinghka, E. Jonas, C. Petschulat, B. Cronin, P. Shafto, and J. B. Tenenbaum. Cross-categorization: A method for discovering multiple overlapping clusterings. In *Proc. of Nonparametric Bayes Workshop at NIPS 2009*, 2009.

[58] A. McCallum. A note on the unification of information extraction and data mining using conditional-probability, relational models. In *In Proceedings of the IJCAI-2003 Workshop on Learning Statistical Models from Relational Data*, 2003.

[59] D. McCarthy and R. Navigli. SemEval-2007 task 10: English lexical substitution task. In *SemEval '07: Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 48–53, Morristown, NJ, USA, 2007. Association for Computational Linguistics.

[60] K. McRae, G. S. Cree, M. S. Seidenberg, and C. McNorgan. Semantic feature production norms for a large set of living and nonliving things. *Behavioral Research Methods*, 37(4):547–559, 2005.

[61] R. Mihalcea, C. Corley, and C. Strapparava. Corpus-based and knowledge-based measures of text semantic similarity. In *In Proceedings of AAAI 2006*, pages 775–780, 2006.

[62] G. A. Miller and W. G. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991.

[63] G. L. Murphy. *The Big Book of Concepts*. The MIT Press, 2002.

[64] R. M. Neal. Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9(2):249–265, 2000.

[65] M. Paşca. Turning Web text and search queries into factual knowledge: Hierarchical class attribute extraction. In *Proceedings of the 23rd National Conference on Artificial Intelligence (AAAI-08)*, pages 1225–1230, Chicago, Illinois, 2008.

[66] M. Paşca and E. Alfonseca. Web-derived resources for Web Information Retrieval: From conceptual hierarchies to attribute hierarchies. In *Proceedings of the 32nd International Conference on Research and Development in Information Retrieval (SIGIR-09)*, Boston, Massachusetts, 2009.

[67] M. Paşca and B. Van Durme. Weakly-supervised acquisition of open-domain classes and class attributes from web documents and query logs. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL-08)*, pages 19–27, Columbus, Ohio, 2008.

[68] S. Padó and M. Lapata. Dependency-based construction of semantic space models. *Comput. Linguist.*, 33(2):161–199, 2007.

[69] S. Padó, U. Padó, and K. Erk. Flexible, corpus-based modelling of human plausibility judgements. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 400–409, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[70] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 79–86, 2002.

[71] P. Pantel and D. Lin. Discovering word senses from text. In *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 613–619, New York, NY, USA, 2002. ACM.

[72] P. Pantel and M. Pennacchiotti. Espresso: leveraging generic patterns for automatically harvesting semantic relations. In *ACL '06: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*, pages 113–120, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

[73] P. Pantel, R. Bhagat, T. Chklovski, and E. Hovy. ISP: Learning inferential selectional preferences. In *In Proceedings of NAACL 2007*, 2007.

[74] P. A. Pantel. *Clustering by committee*. PhD thesis, Edmonton, Alta., Canada, 2003.

[75] L. Parsons, E. Haque, and H. Liu. Subspace clustering for high dimensional data: A review. *SIGKDD Explor. Newsl.*, 6(1):90–105, June 2004.

[76] F. Pereira, N. Tishby, and L. Lee. Distributional clustering of english words. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL-93)*, pages 183–190, Columbus, Ohio, 1993.

[77] F. C. N. Pereira, N. Tishby, and L. Lee. Distributional clustering of English words. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics (ACL-93)*, pages 183–190, Columbus, Ohio, 1993.

[78] X.-H. Phan, L.-M. Nguyen, and S. Horiguchi. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 91–100, New York, NY, USA, 2008. ACM.

[79] J. Pitman. Exchangeable and partially exchangeable random partitions. *Probab. Theory Related Fields*, 102(2):145–158, 1995.

[80] J. Pitman and M. Yor. The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25(2):855–900, 1997.

[81] M. Poesio, R. Mehta, A. Maroudas, and J. Hitzeman. Learning to resolve bridging references. In *ACL '04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 143, Morristown, NJ, USA, 2004. Association for Computational Linguistics.

[82] H. Poon and P. Domingos. Unsupervised semantic parsing. In *Proc. of EMNLP 2009*. Association for Computational Linguistics, 2009.

[83] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 248–256, Singapore, Aug. 2009. Association for Computational Linguistics.

[84] D. Ramage, A. N. Rafferty, and C. D. Manning. Random walks for text semantic similarity. In *Proc. of the 2009 Workshop on Graph-based Methods for Natural Language Processing (TextGraphs-4)*, pages 23–31, 2009.

[85] D. Ramage, S. Dumais, and D. Liebling. Characterizing microblogs with topic models. In *ICWSM*, 2010.

[86] C. E. Rasmussen. The infinite Gaussian mixture model. In *Advances in Neural Information Processing Systems*, pages 554–560. MIT Press, 2000.

[87] J. Reisinger. Extracting salient properties using hierarchical latent class models, In submission.

[88] J. Reisinger and R. Mooney. Multi-prototype vector-space models of word meaning. In *Proc. of NAACL 2010*. Association for Computational Linguistics, 2010.

[89] J. Reisinger and M. Paşca. Latent variable models of concept-attribute attachment. In *Proc. of ACL 2009*, pages 620–628. Association for Computational Linguistics, 2009.

[90] A. Rényi. On measures of information and entropy. In *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability*, page 547561, 1960.

[91] P. Resnik. Selectional preference and sense disambiguation. In *Proceedings of ACL SIGLEX Workshop on Tagging Text with Lexical Semantics*, pages 52–57, Washington, D.C., 1997. ACL.

[92] A. Ritter, Mausam, and O. Etzioni. A latent dirichlet allocation method for selectional preferences. In *In Proceedings of ACL 2010*, 2010.

[93] B. H. Ross and G. L. Murphy. Food for thought: Cross-classification and category organization in a complex real-world domain. *Cognitive Psychology*, 38:495–553, 1999.

[94] Y. Rosseel. Mixture models of categorization. *J. Math. Psychol.*, 46(2):178–210, 2002.

[95] V. Roth and T. Lange. Feature selection in clustering problems. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2004.

[96] M. Sanderson. Word sense disambiguation and information retrieval. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 142–151, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[97] R. Sasano and S. Kurohashi. A probabilistic model for associative anaphora resolution. In *EMNLP '09: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1455–1464, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[98] H. Schütze. Automatic word sense discrimination. *Computational Linguistics*, 24 (1):97–123, 1998.

[99] P. Shafto and J. D. Coley. Development of categorization and reasoning in the natural world: Novices to experts, naive similarity to ecological knowledge. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 2003.

[100] P. Shafto, C. Kemp, V. Mansinghka, M. Gordon, and J. B. Tenenbaum. Learning cross-cutting systems of categories. In *Proc. CogSci 2006*, 2006.

[101] H. Shan and A. Banerjee. Residual Bayesian co-clustering for matrix approximation. In *SIAM International Conference on Data Mining (SDM) 2010*, 2010.

[102] J. Sivic, B. Russell, A. Zisserman, W. Freeman, and A. Efros. Unsupervised discovery of visual object class hierarchies. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR-08)*, pages 1–8, Anchorage, Alaska, 2008.

[103] R. Snow, O'Connor, D. Jurafsky, and A. Ng. Cheap and fast—but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of Empirical Methods in Natural Language Processing*, 2008.

[104] A. Strehl and J. Ghosh. Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3:583–617, 2003.

[105] F. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge unifying WordNet and Wikipedia. In *Proceedings of the 16th World Wide Web Conference (WWW-07)*, pages 697–706, Banff, Canada, 2007.

[106] M. Tatu and D. Moldovan. A semantic approach to recognizing textual entailment. In *Proc. of HLT-EMNLP 2005*. Association for Computational Linguistics, 2005.

[107] Y. W. Teh, H. Daumé III, and D. Roy. Bayesian agglomerative clustering with coalescents. In *Proceedings of the Conference on Neural Information Processing Systems (NIPS)*, Vancouver, Canada, 2007.

[108] K. Tokunaga, J. Kazama, and K. Torisawa. Automatic discovery of attribute words from Web documents. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP-05)*, pages 106–118, Jeju Island, Korea, 2005.

[109] P. D. Turney. Similarity of semantic relations. *Computational Linguistics*, 32(3): 379–416, 2006.

[110] A. Tversky and I. Gati. Similarity, separability, and the triangle inequality. *Psychological Review*, 89(2):123–154, 1982.

[111] B. Van Durme and M. Paşca. Finding cars, goddesses and enzymes: Parametrizable acquisition of labeled instances for open-domain information extraction. In *Proc. of AAAI 2008*, 2008.

[112] B. Vandekerckhove, D. Sandra, and W. Daelemans. A robust and extensible exemplar-based model of thematic fit. In *Proc. of EACL 2009*, pages 826–834. Association for Computational Linguistics, 2009.

[113] I. Verdinelli and L. Wasserman. Bayesian analysis of outlier problems using the Gibbs sampler. *Statistics and Computing*, 1(2), 1991.

[114] E. Voorhees and D. Tice. Building a question-answering test collection. In *Proceedings of the 23rd International Conference on Research and Development in Information Retrieval (SIGIR-00)*, pages 200–207, Athens, Greece, 2000.

[115] W. Voorspoels, W. Vanpaemel, and G. Storms. The role of extensional information in conceptual combination. In *Proceedings of the 31th Annual Conference of the Cognitive Science Society*, 2009.

[116] H. Wallach, D. Mimno, and A. McCallum. Rethinking LDA: Why priors matter. In *NIPS*. 2009.

[117] A. Wierzbicka. *Cross-cultural pragmatics : The semantics of human interaction*. Mouton de Gruyter, Berlin ; New York, 1991.

[118] S. Williamson, C. Wang, K. A. Heller, and D. M. Blei. The IBP-compound Dirichlet process and its application to focused topic modeling. In *Proceedings of the 27th International Conference on Machine Learning*. 2010.

[119] F. Wu and D. Weld. Automatically refining the Wikipedia infobox ontology. In *Proceedings of the 17th World Wide Web Conference (WWW-08)*, pages 635–644, Beijing, China, 2008.

[120] F. Wu and D. Weld. Automatically refining the wikipedia infobox ontology. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 635–644, New York, NY, USA, 2008. ACM.

[121] N. Xue, J. Chen, and M. Palmer. Aligning features with sense distinction dimensions. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 921–928, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

[122] N. Yoshinaga and K. Torisawa. Open-domain attribute-value acquisition from semi-structured texts. In *Proc. of the 6th International Semantic Web Conference (ISWC-07), Workshop on Text to Knowledge: The Lexicon/Ontology Interface (OntoLex-2007)*, pages 55–66, 2007.

[123] G. Zipf. *The Psycho-Biology of Language*. Houghton Mifflin, Boston, MA, 1935.