Cross-Cutting Models of Lexical Semantics

Joseph Reisinger
Department of Computer Sciences
The University of Texas at Austin
Austin, TX 78712
joeraii@cs.utexas.edu

Raymond Mooney
Department of Computer Sciences
The University of Texas at Austin
Austin, TX 78712
mooney@cs.utexas.edu

Abstract

Context-dependent word similarity can be measured over multiple *cross-cutting* dimensions. For example, *lung* and *breath* are similar thematically, while *authoritative* and *superficial* occur in similar syntactic contexts, but share little semantic similarity. Both of these notions of similarity play a role in determining word meaning, and hence lexical semantic models must take them both into account. Towards this end, we develop a novel model, Multi-View Mixture (MVM), that represents words as multiple *overlapping clusterings*. MVM finds multiple data partitions based on different subsets of features, subject to the marginal constraint that feature subsets are distributed according to Latent Dirichlet Allocation. Intuitively, this constraint favors feature partitions that have coherent topical semantics. Furthermore, MVM uses soft feature assignment, hence the contribution of each data point to each clustering view is variable, isolating the impact of data only to views where they assign the most features. Through a series of experiments, we demonstrate the utility of MVM as an inductive bias for capturing relations between words that are intuitive to humans, outperforming related models such as Latent Dirichlet Allocation.

1 Introduction

Humans categorize objects using multiple orthogonal taxonomic systems, where category generalization depends critically on what features are relevant to one particular system. For example, foods can be organized in terms of their nutritional value (high in fiber) or situationally (commonly eaten for Thanksgiving; Shafto et al. (2006)). Human knowledgebases such as Wikipedia also exhibit such multiple clustering structure (e.g. people are organized by occupation or by nationality). The effects of these overlapping categorization systems manifest themselves at the lexical semantic level (Murphy, 2002), 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.

In this work, we introduce a novel probabilistic clustering method, Multi-View Mixture (MVM), based on *cross-cutting categorization* (Shafto et al., 2006) that generalizes traditional *vector-space* or *distributional* models of lexical semantics (Curran, 2004; Padó and Lapata, 2007; Schütze, 1998; Turney, 2006). 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 both *syntagmatic* and *paradigmatic* notions of word relatedness, breaking up word features into multiple categorization systems and then computing similarity separately for each system.

MVM leverages primitives from Dirichlet-Process Mixture Models (DPMMs) and Latent Dirichlet Allocation (LDA). Each clustering (*view*) in MVM consists of a distribution over features and data and views are further subdivided into clusters based on a DPMM. View marginal distributions are determined by LDA, allowing data features to be distributed over multiple views, explaining subsets of features.

Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 1405–1415,
Edinburgh, Scotland, UK, July 27–31, 2011.

We evaluate MVM against several other model-based clustering procedures in a series of human evaluation tasks, measuring its ability to find meaningful syntagmatic and paradigmatic structure. We find that MVM finds more semantically and syntactically coherent fine-grained structure, using both common and rare n-gram contexts.

2 Mixture Modeling and Lexical Semantics

Distributional, or *vector space* methods attempt to model word meaning by embedding words in a common metric space, whose dimensions are derived from, e.g., word collocations (Schütze, 1998), syntactic relations (Padó and Lapata, 2007), or latent semantic spaces (Finkelstein et al., 2001; Landauer and Dumais, 1997; Turian et al., 2010). The distributional hypothesis addresses the problem of modeling word similarity (Curran, 2004; Miller and Charles, 1991; Schütze, 1998; Turney, 2006), and can be extended to selectional preference (Resnik, 1997) and lexical substitution (McCarthy and Navigli, 2007) as well. Such methods are highly scalable (Gorman and Curran, 2006) and have been applied in information retrieval (Manning et al., 2008), large-scale taxonomy induction (Snow et al., 2006), and knowledge acquisition (Van Durme and Paşca, 2008).

Vector space models fail to capture the richness of word meaning since similarity is not a globally consistent metric. It violates, e.g., the triangle inequality: the sum of distances from *bat* to *club* and *club* to *association* is less than the distance from *bat* to *association* (Griffiths et al., 2007; Tversky and Gati, 1982).[1] Erk (2007) circumvents this problem by representing words as multiple exemplars derived directly from word occurrences and embedded in a common vector space to capture context-dependent usage. Likewise Reisinger and Mooney (2010) take a similar approach using mixture modeling combined with a background variation model to generate multiple prototype vectors for polysemous words.

Both of these approaches still ultimately embed all words in a single metric space and hence argue for globally consistent metrics that capture human intuitive notions of "similarity." Rather than assuming a global metric embedding exists, in this work we simply leverage the *cluster assumption*, e.g. that similar words should appear in the same clusters, in particular extending it to multiple clusterings. The cluster assumption is a natural fit for lexical semantics, as partitions can account for metric violations. The end result is a model capable of representing multiple, overlapping similarity metrics that result in disparate valid clusterings leveraging the

> **Subspace Hypothesis**: For any pair of words, the set of "active" features governing their apparent similarity differs. 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 occurring in similar documents.

MVM can extract multiple competing notions of similarity, for example both *paradigmatic*, or thematic similarity, and *syntagmatic* or syntactic similarity, in addition to more fine grained relations.

3 Multi-View Clustering with MVM

As feature dimensionality increases, the number of ways the data can exhibit interesting structure goes up exponentially. Clustering is commonly used to explain data, but often there are several equally valid, competing clusterings, keying off of different subsets of features, especially in high-dimensional settings such as text mining (Niu et al., 2010). For example, company websites can be clustered by sector or by geographic location, with one particular clustering becoming predominant when a majority of features correlate with it. In fact, informative features in one clustering may be noise in another, e.g. the occurrence of *CEO* is not necessarily discriminative when clustering companies by industry sector, but may be useful in other clusterings. Multiple clustering is one approach to inferring feature subspaces that lead to high quality data partitions. Multiple clustering also improves the flexibility of generative clustering models, as a single model is no longer required to explain all the variance in the feature dimensions (Mansinghka et al., 2009).

[1] Similarity also has been shown to violate symmetry (e.g. people have the intuition that *China* is more similar to *North Korea* than *North Korea* is to *China*).

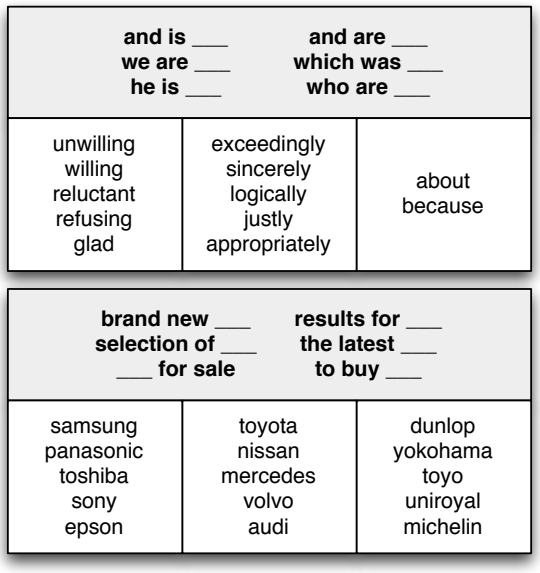


Figure 1: Example clusterings from MVM applied to Google n-gram data. Top contexts (features) for each view are shown, along with examples of word clusters. Although these particular examples are interpretable, in general the relationship captured by the view's context subspace is not easily summarized.

MVM is a multinomial-Dirichlet multiple clustering procedure for distributional lexical semantics that fits multiple, overlapping Dirichlet Process Mixture Models (DPMM) to a set of word data. Features are distributed across the set of clusterings (views) using LDA, and each DPMM is fit using a subset of the features. This reduces clustering noise and allows MVM to capture multiple ways in which the data can be partitioned. Figure 1 shows a simple example, and Figure 2 shows a larger sample of feature-view assignments from a 3-view MVM fit to contexts drawn from the Google n-gram corpus.

We implement MVM using generative model primitives drawn from Latent Dirichlet Allocation (LDA) and the Dirichlet Process (DP). $|M|$ disparate clusterings (views) are inferred jointly from a set of data $\mathcal{D} = \{\mathbf{w}_d | d \in [1 \dots D]\}$. Each data vector $\mathbf{w}_d$ is associated with a probability distribution over views $\boldsymbol{\theta}_d^{|M|}$. Empirically, $\boldsymbol{\theta}_d^{|M|}$ is represented as a set of *feature-view* assignments $\mathbf{z}_d$, sampled via the standard LDA collapsed Gibbs sampler. Hence, each view maintains a separate distribution over features. The generative model for feature-view assignment is given by

$$\begin{aligned}
\boldsymbol{\theta}_d^{|M|} | \boldsymbol{\alpha} &\sim \text{Dirichlet}(\boldsymbol{\alpha}), & d &\in D, \\
\boldsymbol{\phi}_m | \boldsymbol{\beta} &\sim \text{Dirichlet}(\boldsymbol{\beta}), & m &\in |M|, \\
z_{dn} | \boldsymbol{\theta}_d &\sim \text{Discrete}(\boldsymbol{\theta}_d), & n &\in |\mathbf{w}_d|, \\
w_{dn} | \boldsymbol{\phi}_{z_{dn} m} &\sim \text{Discrete}(\boldsymbol{\phi}_{z_{dn} m}), & n &\in |\mathbf{w}_d|,
\end{aligned}$$

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

Conditional on the feature-view assignment $\{\mathbf{z}\}$, a clustering is inferred for each view using the Chinese Restaurant Process representation of the DP. The clustering probability is given by

$$\begin{aligned}
p(\mathbf{c} | \mathbf{z}, \mathbf{w}) &\propto p(\{\mathbf{c}_m\}, \mathbf{z}, \mathbf{w}) \\
&= \prod_{m=1}^{M} \prod_{d=1}^{|D|} p(\mathbf{w}_d^{[\mathbf{z}=m]} | \mathbf{c}_m, \mathbf{z}) p(\mathbf{c}_m | \mathbf{z}).
\end{aligned}$$

where $p(\mathbf{c}_m | \mathbf{z})$ is a prior on the clustering for view m, i.e. the DPMM, and $p(\mathbf{w}_d^{[\mathbf{z}=m]} | \mathbf{c}_m, \mathbf{z})$ is the likelihood of the clustering $\mathbf{c}_m$ given the data point $\mathbf{w}_d$ restricted to the features assigned to view m:

$$\mathbf{w}_d^{[\mathbf{z}=m]} \stackrel{\text{def}}{=} \{w_{id} | z_{id} = m\}.$$

Thus, we treat the m clusterings $\mathbf{c}_m$ as conditionally independent given the feature-view assignments.

The feature-view assignments $\{\mathbf{z}\}$ act as a set of marginal constraints on the multiple clusterings, and the impact that each data point can have on each clustering is limited by the number of features assigned to it. For example, in a two-view model, $z_{id} = 1$ might be set for all syntactic features (yielding a syntagmatic clustering) while $z_{id} = 2$ is set for document features (paradigmatic clustering).

By allowing the clustering model capacity to vary via the DPMM, MVM can naturally account for the semantic variance of the view. This provides a novel mechanism for handling feature noise: noisy features can be assigned to a separate view with potentially a small number of clusters. This phenomenon is apparent in cluster 1, view 1 in the example in figure 2, where place names and adjectives are clustered together using rare contexts

From a topic modeling perspective, MVM finds topic refinements within each view, similar to hierarchical methods such as the nested Chinese Restaurant Process (Blei et al., 2003). The main difference is that the features assigned to the second "refined topics" level are constrained by the higher

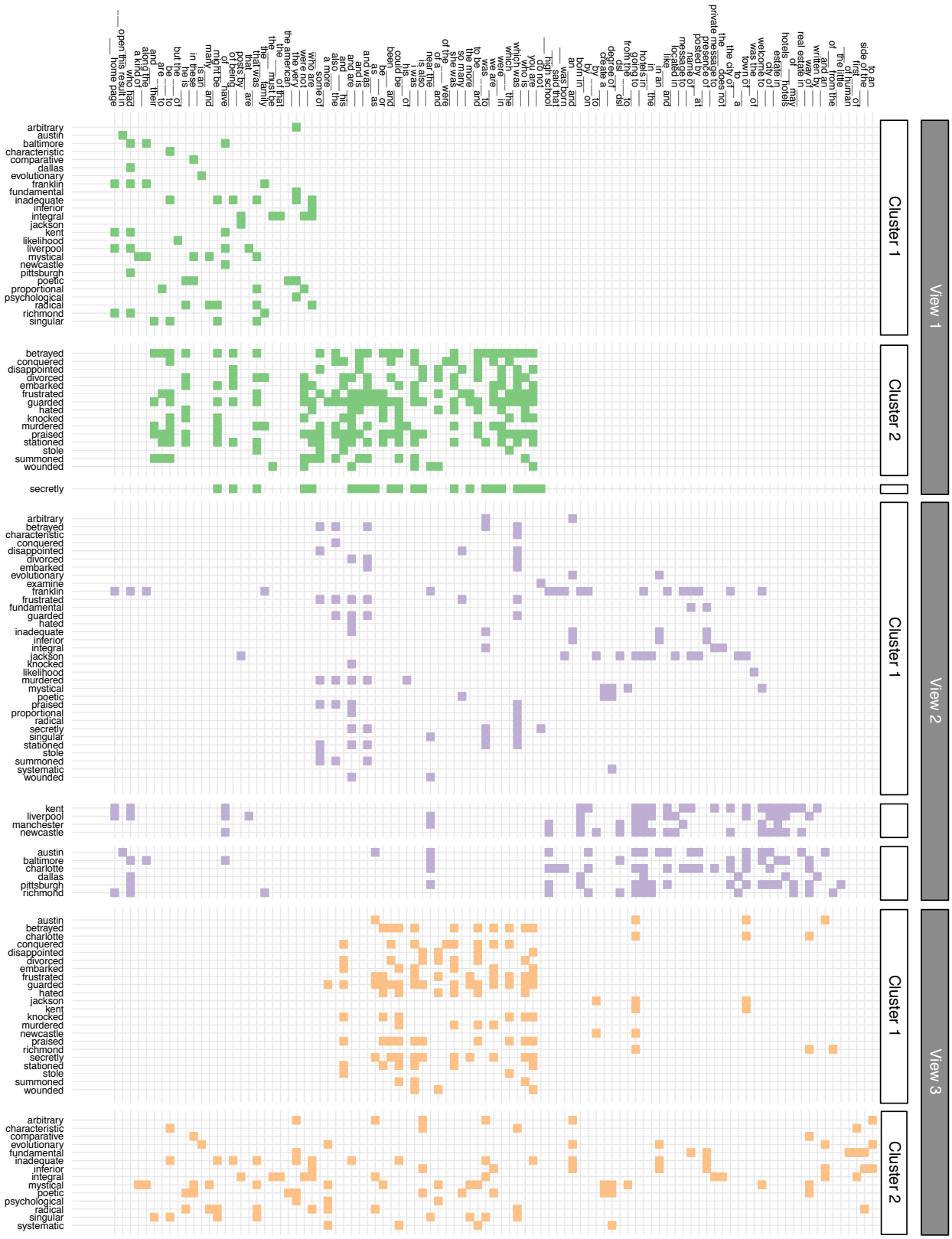


Figure 2: **Topics with Senses**: Shows top 20% of features for each view in a 3-view MVM fit to Google n-gram context data; different views place different mass on different sets of features. Cluster groupings within each view are shown. View 1 cluster 2 and View 3 cluster 1 both contain past-tense verbs, but only overlap on a subset of syntactic features.

level, similar to hierarchical clustering. Unlike hierarchical clustering, however, the top level topics/views form an admixture, allowing individual features from a single data point to be assigned to multiple views.

The most similar model to ours is *Cross-cutting categorization* (CCC), which fits multiple DPMMs to non-overlapping partitions of features (Mansinghka et al., 2009; Shafto et al., 2006). Unlike MVM, CCC *partitions* features among multiple DPMMs, hence all occurrences of a particular feature will end up in a single clustering, instead of assigning them softly using LDA. Such hard feature partitioning does not admit an efficient sampling procedure, and hence Shafto et al. (2006) rely on Metropolis-Hastings steps to perform feature assignment, making the model less scalable.

3.1 Word Representation

MVM is trained as a lexical semantic model on Web-scale n-gram and semantic context data. N-gram contexts are drawn from a combination of the Google n-gram and Google books n-gram corpora, with the head word removed: e.g. for the term *architect*, we collect contexts such as *the __ of the house*, *an __ is a*, and *the __ of the universe*. Semantic contexts are derived from word occurrence in Wikipedia documents: each document a word appears in is added as a potential feature for that word. This co-occurrence matrix is the transpose of the standard bag-of-words document representation.

In this paper we focus on two representations:

1. **Syntax-only** – Words are represented as bags of ngram contexts derived slot-filling procedure described above.

2. **Syntax+Documents** – The syntax-only representation is augmented with additional document contexts drawn from Wikipedia.

Models trained on the **syntax-only** set are only capable of capturing *syntagmatic* similarity relations, that is, words that tend to appear in similar contexts. In contrast, the **syntax+documents** set broadens the scope of modelable similarity relations, allowing for *paradigmatic* similarity (e.g. words that are topically related, but do not necessarily share common syntactic contexts).

Given such word representation data, MVM generates a fixed set of M context views corresponding to dominant eigenvectors in local syntactic or semantic space. Within each view, MVM partitions words into clusters based on each word's *local representation* in that view; that is, based on the set of context features it allocates to the view. Words have a non-uniform affinity for each view, and hence may not be present in every clustering (Figure 2). This is important as different ways of drawing distinctions between words do not necessarily apply to all words. In contrast, LDA finds locally consistent collections of contexts but does not further subdivide words into clusters given that set of contexts. Hence, it may miss more fine-grained structure, even with increased model complexity.

4 Experimental Setup

4.1 Corpora

We derive word features from three corpora: (1) the English **Google Web n-gram** corpus, containing n-gram contexts up to 5-gram that occur more than 40 times in a 1T word corpus of Web text, (2) the English **Google Books n-gram** corpus[2], consisting of n-gram contexts up to 5-gram that occur more than 40 times in a 500B word corpus of books, and (3) a snapshot of the English Wikipedia[3] taken on October 11, 2010 containing over 3M articles.

MVM is trained on a sample of 20k English words drawn uniformly at random from the top 200k English terms appearing in Wikipedia (different parts of speech were sampled from the Google n-gram corpus according to their observed frequency). Two versions of the **syntax-only** dataset are created from different subsets of the Google n-gram corpora: (1) the *common* subset contains all syntactic contexts appearing more than 200 times in the combined corpus, and (2) the *rare* subset, containing only contexts that appear 50 times or fewer.

4.2 Human Evaluation

Our main goal in this work is to find models that capture aspects of the syntactic and semantic organization of word in text that are intuitive to humans.

[2] http://ngrams.googlelabs.com/datasets
[3] http://wikipedia.org

Context Intrusion			Word Intrusion		
__ is characterized	top of the __	*country to* __	metal	dues	humor
symptoms of __	*of __ understood*	__ or less	floral	premiums	ingenuity
cases of __	along the __	__ a year	nylon	pensions	*advertisers*
in cases of __	portion of the __	__ per day	*what*	*did*	delight
real estate in __	side of the __	__ or more	ruby	damages	astonishment

Document Intrusion		
Puerto Rican cuisine	Adolf Hitler	History of the Han Dynasty
Greek cuisine	*List of General Hospital characters*	Romance of the Three Kingdoms
ThinkPad	History of France	*List of dog diseases*
Palestinian cuisine	Joachim von Ribbentrop	Conquest of Wu by Jin
Field ration	World War I	Mongolia

Table 1: Example questions from the three intrusion tasks, in order of difficulty (left to right, easy to hard; computed from inter-annotator agreement). *Italics* show intruder items.

According to the *use theory* of meaning, lexical semantic knowledge is equivalent to knowing the contexts that words appear in, and hence being able to form reasonable hypotheses about the relatedness of syntactic contexts.

Vector space models are commonly evaluated by comparing their similarity predictions to a nominal set of human similarity judgments (Curran, 2004; Padó and Lapata, 2007; Schütze, 1998; Turney, 2006). In this work, since we are evaluating models that potentially yield many different similarity scores, we take a different approach, scoring clusters on their semantic and syntactic *coherence* using a *set intrusion* task (Chang et al., 2009).

In set intrusion, human raters are shown a set of options from a coherent group and asked to identify a single *intruder* drawn from a different group. We extend intrusion to three different lexical semantic tasks: (1) *context intrusion*, where the top contexts from each cluster are used, (3) *document intrusion*, where the top document contexts from each cluster are used, and (2) *word intrusion*, where the top words from each cluster are used. For each cluster, the top four contexts/words are selected and appended with another context/word from a different cluster.[4] The resulting set is then shuffled, and the human raters are asked to identify the intruder, after being given a short introduction (with common examples) to the task. Table 1 shows sample questions of varying degrees of difficulty. As the semantic coherence and distinctness from other clusters increases, this task becomes easier.

Set intrusion is a more robust way to account for human similarity judgments than asking directly for a numeric score (e.g., the Miller and Charles (1991) set) as less calibration is required across raters. Furthermore, the additional cluster context significantly reduces the variability of responses.

Human raters were recruited from *Amazon's Mechanical Turk*. A total of 1256 raters completed 30438 evaluations for 5780 unique intrusion tasks (5 evaluations per task). 2736 potentially fraudulent evaluations from 11 raters were rejected.[5] Table 3 summarizes inter-annotator agreement. Overall we found $\kappa \approx 0.4$ for most tasks; a set of comments about the task difficulty is given in Table 2, drawn from an anonymous public message board.

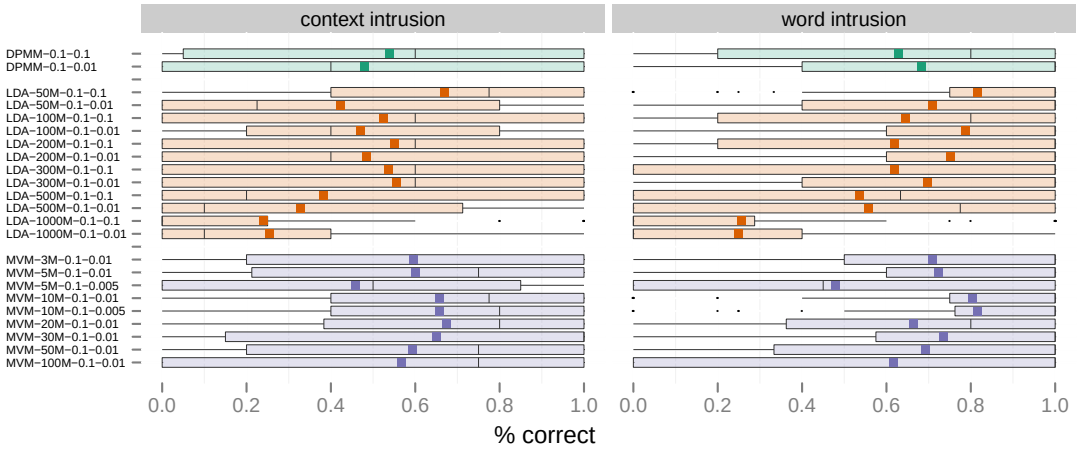
5 Results

We trained DPMM, LDA and MVM models on the **syntax-only** and **syntax+documents** data across a wide range of settings for $M \in \{3, 5, 7, 10, 20, 30, 50, 100, 200, 300, 500, 1000\}$,[6]

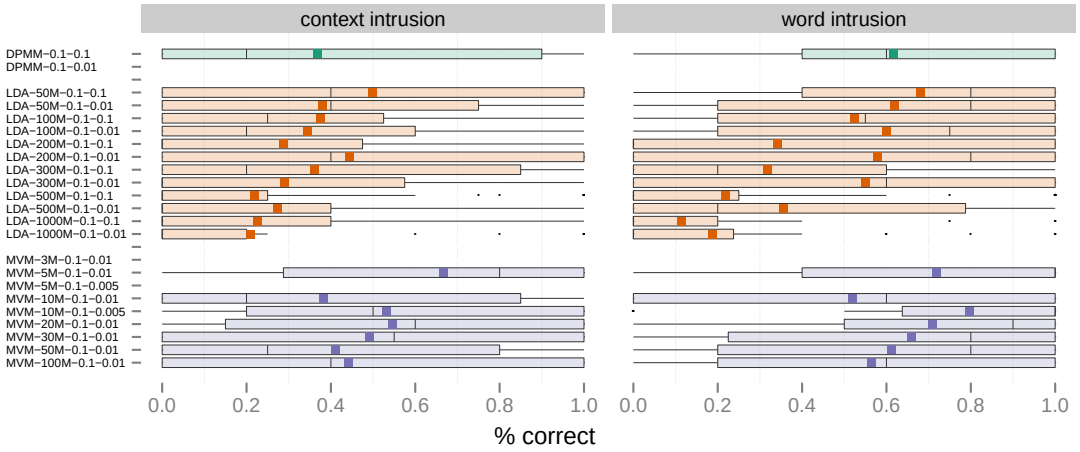
[4] Choosing four elements from the cluster uniformly at random instead of the top by probability led to lower performance across all models.

[5] (**Rater Quality**) Fraudulent Turkers were identified using a combination of average answer time, answer entropy, average agreement with other raters, and adjusted answer accuracy.

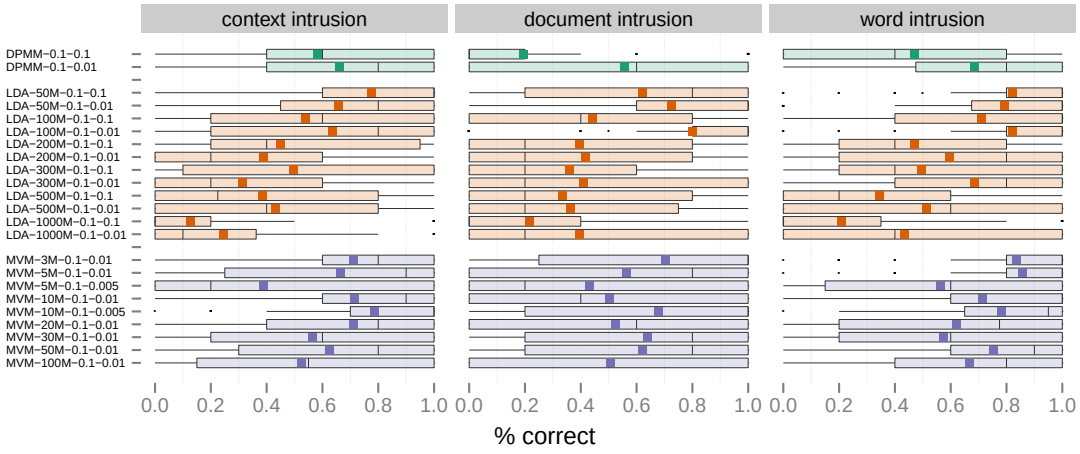
[6] LDA is run on a different range of M settings from MVM (50-1000 vs 3-100) in order to keep the effective number of



(a) **Syntax-only**, common n-gram contexts.



(b) **Syntax-only**, rare n-gram contexts.



(c) **Syntax+Documents**, common n-gram contexts.

Figure 3: Average scores for each model broken down by parameterization and data source. Error bars depict 95% confidence intervals. X-axis labels show **Model-views**-α-β. Dots show average rater scores; bar-charts show standard quantile ranges and median score.

U1	I just tried 30 of the what doesn't belong ones. They took about 30 seconds each due to thinking time so not worth it for me.
U2	I don't understand the fill in the blank ones to be honest. I just kinda pick one,since I don't know what's expected lol
U3	Your not filling in the blank just ignore the blank and think about how the words they show relate to each other and choose the one that relates least. Some have just words and no blanks.
U4	These seem very subjective to mw. i hope there isn't definite correct answers because some of them make me go [emoticon of head-scratching]
U5	I looked and have no idea. I guess I'm a word idiot because I don't see the relation between the words in the preview HIT - too scared to try any of these.
U6	I didn't dive in but I did more than I should have they were just too easy. Most of them I could tell what did not belong, some were pretty iffy though.

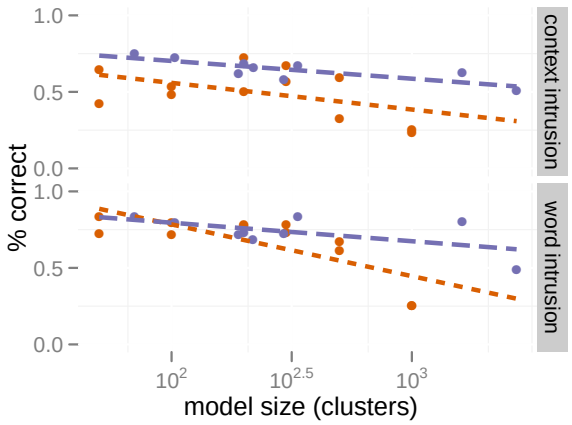
Table 2: Sample of comments about the task taken verbatim from a public Mechanical Turk user message board (TurkerNation). Overall the raters report the task to be difficult, but engaging.

$\alpha \in \{0.1, 0.01\}$, and $\beta \in \{0.1, 0.05, 0.01\}$ in order to understand how they perform relatively on the intrusion tasks and also how sensitive they are to various parameter settings.[7] Models were run until convergence, defined as no increase in log-likelihood on the training set for 100 Gibbs samples. Average runtimes varied from a few hours to a few days, depending on the number of clusters or topics. There is little computational overhead for MVM compared to LDA or DPMM with a similar number of clusters.

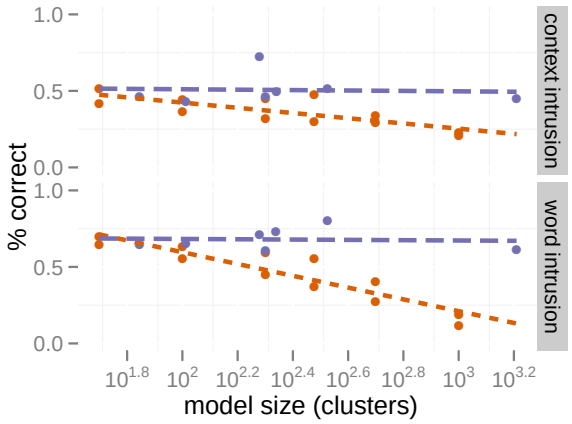
Overall, MVM significantly outperforms both LDA and DPMM (measured as % of intruders correctly identified) as the number of clusters increases. Coarse-grained lexical semantic distinctions are easy for humans to make, and hence models with fewer clusters tend to outperform models with more clusters. Since high granularity predictions are more useful for downstream tasks, we focus on the interplay between model complexity and performance.

clusters (and hence model capacity) roughly comparable.

[7]We did not compare directly to Cross-cutting categorization, as the Metropolis-Hasting steps required that model were too prohibitively expensive to scale to the Google n-gram data.



(a) **Syntax-only**, common n-gram contexts.



(b) **Syntax-only**, rare n-gram contexts.

Figure 4: Scatterplot of model size vs. avg score for MVM (dashed, purple) and LDA (dotted, orange).

5.1 Syntax-only Model

For common n-gram context features, MVM performance is significantly less variable than LDA on both the word intrusion and context intrusion tasks, and furthermore significantly outperforms DPMM (Figure 3(a)). For context intrusion, DPMM, LDA, and MVM average 57.4%, 49.5% and 64.5% accuracy respectively; for word intrusion, DPMM, LDA, and MVM average 66.7%, 66.1% and 73.6% accuracy respectively (averaged over all parameter settings). These models vary significantly in the average number of clusters used: 373.5 for DPMM, 358.3 for LDA and 639.8 for MVM, i.e. the MVM model is signifi-

Model	Syntax	Syntax+Documents	Overall
DPMM	0.30	0.40	**0.33**
LDA	0.33	0.39	**0.35**
MVM	0.44	0.49	**0.46**
Overall	**0.37**	**0.43**	**0.39**

Table 3: Fleiss' κ scores for various model and data combinations. Results from MVM have higher κ scores than LDA or DPMM; likewise **Syntax+Documents** data yields higher agreement, primarily due to the relative ease of the document intrusion task.

cantly more granular. Figure 4(a) breaks out model performance by model complexity, demonstrating that MVM has a significant edge over LDA as model complexity increases.

For rare n-gram contexts, we obtain similar results, with MVM scores being less variable across model parameterizations and complexity (Figure 3(b)). In general, LDA performance degrades faster as model complexity increases for rare contexts, due to the increased data sparsity (Figure 4(b)). For context intrusion, DPMM, LDA, and MVM average 45.9%, 36.1% and 50.9% accuracy respectively; for word intrusion, DPMM, LDA, and MVM average 67.4%, 45.6% and 67.9% accuracy; MVM performance does not differ significantly from DPMM, but both outperform LDA. Average cluster sizes are more uniform across model types for rare contexts: 384.0 for DPMM, 358.3 for LDA and 391 for MVM.

Human performance on the context intrusion task is significantly more variable than on the word-intrusion task, reflecting the additional complexity.

In all models, there is a high correlation between rater scores and per-cluster likelihood, indicating that model confidence reflects noise in the data.

5.2 Syntax+Documents Model

With the **syntax+documents** training set, MVM significantly outperforms LDA across a wide range of model settings. MVM also outperforms DPMM for word and document intrusion. For context intrusion, DPMM, LDA, and MVM average 68.0%, 51.3% and 66.9% respectively;[8] for word intrusion, DPMM, LDA, and MVM average 56.3%, 64.0% and 74.9% respectively; for document intrusion, DPMM, LDA,

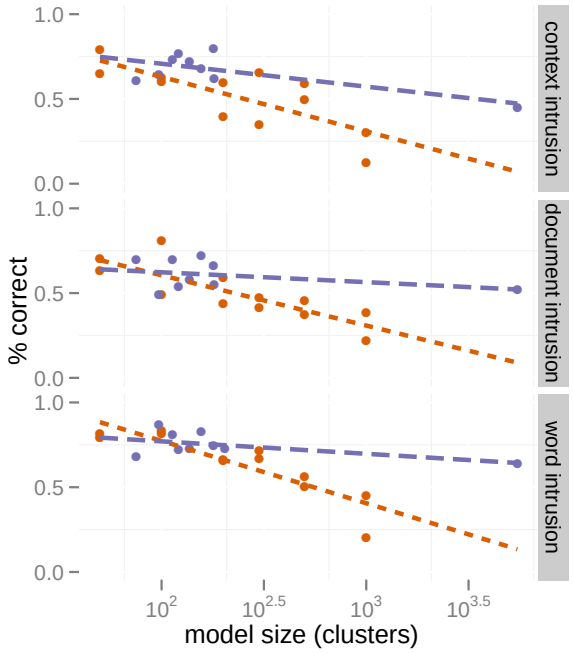


Figure 5: Scatterplot of model size vs. avg score for MVM (dashed, purple) and LDA (dotted, orange); **Syntax+Documents** data.

and MVM average 41.5%, 49.7% and 60.6% respectively. Qualitatively, models trained on **syntax+document** yield a higher degree of paradigmatic clusters which have intuitive thematic structure. Performance on document intrusion is significantly lower and more variable, reflecting the higher degree of world knowledge required. As with the previous data set, performance of MVM models trained on **syntax+documents** data degrades less slowly as the cluster granularity increases (Figure 5).

One interesting question is to what degree MVM *views* partition syntax and document features versus LDA topics. That is, to what degree do the MVM views capture purely syntagmatic or purely paradigmatic variation? We measured *view entropy* for all three models, treating syntactic features and document features as different class labels. MVM with $M = 50$ views obtained an entropy score of 0.045, while LDA with $M = 50$ obtained 0.073, and the best DPMM model 0.082.[9] Thus MVM views may indeed capture pure syntactic or thematic clusterings.

[8] High DPMM accuracy is driven by the low number of clusters: 46.5 for DPMM vs. 358.3 for LDA and 725.6 for MVM.

[9] The low entropy scores reflect the higher percentage of syntactic contexts overall.

5.3 Discussion

As cluster granularity increases, we find that MVM accounts for feature noise better than either LDA or DPMM, yielding more coherent clusters. (Chang et al., 2009) note that LDA performance degrades significantly on a related task as the number of topics increases, reflecting the increasing difficulty for humans in grasping the connection between terms in the same topic. This suggests that as topics become more ne-grained in models with larger number of topics, they are less useful for humans. In this work, we find that although MVM and LDA perform similarity on average, MVM clusters are significantly more interpretable than LDA clusters as the granularity increases (Figures 4 and 5). We argue that models capable of making such fine-grained semantic distinctions are more desirable.

The results presented in the previous two sections hold both for unbiased cluster selection (e.g. where clusters are drawn uniformly at random from the model) *and* when cluster selection is biased based on model probability (results shown). Biased selection potentially gives an advantage to MVM, which generates many more small clusters than either LDA or DPMM, helping it account for noise.

6 Future Work

Models based on cross-cutting categorization is a novel approach to lexical semantics and hence should be evaluated on standard baseline tasks, e.g. contextual paraphrase or *lexical substitution* (McCarthy and Navigli, 2007). Additional areas for future work include:

(**Latent Relation Modeling**) Clusterings formed from feature partitions in MVM can be viewed as a form of *implicit* relation extraction; that is, instead of relying on explicit surface patterns in text, relations between words or concepts are identified indirectly based on common syntactic patterns. For example, clusterings that divide cities by geography or clusterings partition adjectives by their polarity.

(**Latent Semantic Language Modeling**) Generative models such as MVM can be used to build better priors for class-based language modeling (Brown et al., 1992). The rare n-gram results demonstrate that MVM is potentially useful for tail contexts; i.e. inferring tail probabilities from low counts.

(**Explicit Feature Selection**) In this work we rely on smoothing to reduce the noise of over-broad extraction rather than performing feature selection explicitly. All of the models in this paper can be combined with feature selection methods to remove noisy features, and it would be particularly interesting to draw parallels to models of "clutter" in vision.

(**Hierarchical Cross-Categorization**) Human concept organization consists of multiple overlapping local ontologies, similar to the loose ontological structure of Wikipedia. Furthermore, each ontological system has a different set of salient properties. It would be interesting to extend MVM to model hierarchy explicitly, and compare against baselines such as *Brown clustering* (Brown et al., 1992), the nested Chinese Restaurant Process (Blei et al., 2003) and the hierarchical Pachinko Allocation Model (Mimno et al., 2007).

7 Conclusion

This paper introduced MVM, a novel approach to modeling lexical semantic organization using multiple *cross-cutting* clusterings capable of capturing multiple lexical similarity relations jointly in the same model. In addition to robustly handling homonymy and polysemy, MVM naturally captures both *syntagmatic* and *paradigmatic* notions of word similarity. MVM performs favorably compared to other generative lexical semantic models on a set of human evaluations, over a wide range of model settings and textual data sources.

Acknowledgements

We would like to thank the anonymous reviewers for their extensive comments. This work was supported by a Google PhD Fellowship to the first author.

References

David Blei, Thomas Griffiths, Michael Jordan, and Joshua Tenenbaum. 2003. Hierarchical topic models and the nested Chinese restaurant process. In *Proc. NIPS-2003*.

Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18:467–479.

Jonathan Chang, Jordan Boyd-Graber, Chong Wang, Sean Gerrish, and David M. Blei. 2009. Reading tea leaves: How humans interpret topic models. In *NIPS*.

James Curran. 2004. *From Distributional to Semantic Similarity*. Ph.D. thesis, University of Edinburgh.

Katrin Erk. 2007. A simple, similarity-based model for selectional preferences. In *Proc. of the ACL*. Association for Computer Linguistics.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: the concept revisited. In *Proc. of WWW 2001*.

James Gorman and James R. Curran. 2006. Scaling distributional similarity to large corpora. In *Proc. of ACL 2006*.

Thomas L. Griffiths, Mark Steyvers, and Joshua B. Tenenbaum. 2007. Topics in semantic representation. *Psychological Review*, 114:2007.

Thomas Landauer and Susan Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104(2):211–240.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Vikash K. Mansinghka, Eric Jonas, Cap Petschulat, Beau Cronin, Patrick Shafto, and Joshua B. Tenenbaum. 2009. Cross-categorization: A method for discovering multiple overlapping clusterings. In *Proc. of Nonparametric Bayes Workshop at NIPS 2009*.

Diana McCarthy and Roberto Navigli. 2007. SemEval-2007 task 10: English lexical substitution task. In *SemEval '07: Proceedings of the 4th International Workshop on Semantic Evaluations*. Association for Computational Linguistics.

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

David Mimno, Wei Li, and Andrew McCallum. 2007. Mixtures of hierarchical topics with pachinko allocation. In *ICML*.

Gregory L. Murphy. 2002. *The Big Book of Concepts*. The MIT Press.

Donglin Niu, Jennifer G. Dy, and Michael I. Jordan. 2010. Multiple non-redundant spectral clustering views. In Johannes Fürnkranz and Thorsten Joachims, editors, *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 831–838.

Sebastian Padó and Mirella Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

Joseph Reisinger and Raymond J. Mooney. 2010. A mixture model with sharing for lexical semantics. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2010)*.

Philip Resnik. 1997. Selectional preference and sense disambiguation. In *Proceedings of ACL SIGLEX Workshop on Tagging Text with Lexical Semantics*, pages 52–57. ACL.

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123.

Patrick Shafto, Charles Kemp, Vikash Mansinghka, Matthew Gordon, and Joshua B. Tenenbaum. 2006. Learning cross-cutting systems of categories. In *Proc. CogSci 2006*.

Rion Snow, Daniel Jurafsky, and Andrew Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proc. of ACL 2006*.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proc. of the ACL*.

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

Amos Tversky and Itamar Gati. 1982. Similarity, separability, and the triangle inequality. *Psychological Review*, 89(2):123–154.

Benjamin Van Durme and Marius Paşca. 2008. Finding cars, goddesses and enzymes: Parametrizable acquisition of labeled instances for open-domain information extraction. In *Proc. of AAAI 2008*.