Simple Unsupervised Topic Discovery for Attribute Extraction in SEM Tasks using WordNet

Abhimanu Kumar, Richard Chatwin, Joydeep Ghosh

The University of Texas at Austin , Adchemy Inc., The University of Texas at Austin
Department of Computer Science, Adchemy Research Lab, Department of ECE
abhimanu@cs.utexas.edu, richard@adchemy.com, ghosh@ece.utexas.edu

Abstract

We present here a simple approach for topic discovery to extract attributes of online products using Wordnet. Identifying product attributes is important for search engine marketing (SEM) since it is integral to the ads displayed for search queries (Moran and Hunt, 2009). Our wordnet based model provides a simple, scalable and high precision attribute extraction mechanism. It is well suited for identifying attributes for previously unseen product categories and thus works specially well for SEM scenario. It outperforms unsupervised topic discovery approaches such as LDA for SEM tasks on 4 online product datasets. The model has been successfully implemented as a production version code for ad-copy creation.

1. Introduction

Information extraction has been an active area of research in Natural Language Processing. It is useful for obtaining query-able information databases from unstructured data such as webpages, news articles etc. Information extraction approaches has been applied to a variety of tasks from obtaining protein names from biological papers (Fukuda and Tamura, 1998) to building dictionaries (Riloff and Jones, 1998). These techniques have also been used to extract relationship among entities (Zelenko et al., 2003) and entity attribute extractions (Bellare and Talukdar, 2007) using training seed sets.

But all the work so far has focused about: a) finding entities when the entity types are known for ex: finding a person or location from a text, or b) extract entities/relations using a seed set to train the model. This can be problematic for entities hitherto unseen by the model. We propose a simple and scalable information extraction model to discover new entity types without any seed set or prior knowledge of the types to be extracted. This scenario is typically encountered in search engine ad-copy creation process where the attributes of a product being advertised can vary from one product subcategory to another. The seed labels are of not much use in this case. Our proposed model uses WordNet semantic similarity metrics to obtain product attribute sets. The input of the model is online product category catalog and the output is a set of clusters each representing an attribute of the product. This model is well suited for ad creation in SEM tasks and can also be used as a bootstrapping tool for general attribute extraction problems.

The model is unsupervised and doesn't need any seed set for training, though it uses WordNet semantic structure to find the attributes. This makes it highly scalable to newer product categories. The model outputs a specified number, κ, of topics or attribute clusters and ranks them in the decreasing order of confidence. Ad-copy creation process, described in section 2., needs to know the prominence of a product feature and whether to include it in the ad-display. The ranked output of the model helps here in deciding the relevance of an attribute for a given product category. We compare our model with traditional unsupervised topics discovery models such as LDA on 4 SEM datasets. It performs better than LDA on all 4 datasets as reported in later sections. Though the model works well for SEM related tasks and datasets, it is not a generic model like LDA (Blei et. al, 2003). It exploits the unique properties of an SEM task and corresponding datasets and is built for such a task. We discuss the cases when it might perform poorly.

Our attribute extraction model sits at the unique juncture of word-semantics, Ontology, and data statistics based extraction techniques. We combine WordNet based "sense-ontology" and semantic metrics with statistical information present in the data to discover relevant attribute-clusters.

2. Problem Definition

The primary focus of search engine marketing is displaying appropriate ads on search engines for a search term. SEM firms maintain a set of appropriate advertisements related to each search term and choose the best ad from this set based on certain relevance criteria. Table 1 shows a search term "Chaise Lounge" and the corresponding set of candidate ads to be shown. Producing this set of ads is one of the big challenges of SEM. These sets of candidate ads are short sentences made of essentially two parts: a) Intent and b) Noun Phrase. The intent of the ad tells the purpose of the ad, e.g. in "IKEA leather chaise lounge on sale", "on sale" is the intent, and in "buy cheap colorful furniture", "buy" is the intent. The noun phrase is the product being talked about in the ad. In "IKEA leather chaise lounge on sale", "IKEA leather chaise lounge" is the noun phrase, and in "buy cheap colorful furniture", "cheap colorful furniture" is the noun phrase. Noun phrase in the ad is a sequence of the product and its attributes, e.g. "IKEA leather chaise lounge" is made of "IKEA" + "leather" + "chaise lounge". Formally all this can be expressed in terms of a context free grammar as:

$$\langle ad \rangle = (\langle intent \rangle)^{\star} \langle noun\ phrase \rangle (\langle intent \rangle)^{\star}$$
$$\langle noun\ phrase \rangle = (\langle attribute \rangle)^{+} \langle product\ name \rangle \qquad (1)$$

Chaise Lounge		
IKEA leather chaise lounge on sale	affordable home furniture	buy cheap colorful furniture

Table 1: 3 candidate ads for search term “Chaise Lounge”

The $\langle intent \rangle$ is easy to obtain, but finding $\langle attribute \rangle$ set requires domain knowledge. The $\langle attribute \rangle$ of a product helps in defining the specificity of the ad by: a) targeting a specific set of consumers who are interested in that attribute, and b) providing information about the category of products which are available for that $\langle intent \rangle$ at the sellers facilities. E.g. the product toy can have several attributes and the ad “$wooden\ brain-teaser\ puzzles\ for\ sale\ at\ walmart$”, with the help of “wooden” and “brain-teaser” attributes, targets the set of consumers who are interested in wooden brain-teaser puzzles. Knowing this attribute-cluster requires going through the toy catalog of the store and manually extracting these attribute-clusters. For ex: “wooden” attribute is a member of “material” attribute-cluster of toy.

Our model solves this problem by automatically extracting the set of attributes using the seller product catalog. It extracts the attributes as well as provides label to each attribute set. For the product toy mentioned above, the model discovers the attribute-clusters : $\{wooden, leather, plastic, tin \ldots\}$ and $\{red, green, blue, black \ldots\}$ and provides labels “$material$” and “$color$” respectively to these 2 clusters. This helps in ad-copy creation. The ad-copy creation is an extension of the ad-generation scheme in equation 1. The difference is in the $<$ $noun\ phrase$ $>$ generation where the new scheme is:

$$\langle noun\ phrase \rangle = (\langle attribute \rangle_1)?\ \ldots\ (\langle attribute \rangle_n)?\ \langle product\ name \rangle \qquad (2)$$

In the ad-copy equation 2 above, the attributes of the product are assigned certain order to give the ad semantically correct structure. For ex: “coffee-colored women’s t-shirt” is semantically/aesthetically better than “women’s coffee-colored t-shirt”. Knowing attribute-cluster labels makes obtaining the right order among attributes easy.

3. Related Work

A variety of approaches have been used from generative (Freitag and McCallum, 1999) and discriminatory schemes (Yu, Lam and Chen, 2009) to rule based models (Reiss and Raghavan, 2008). Entity and attribute extraction is an important subtask of Information Extraction problem.

Generative and Structure Learning based extraction. (Eisenstein and Yano, 2011) provide a non-parametric generative scheme for named entities extraction from text. It uses supervision from an initial set of 5 prototype examples. (Reisinger and Pasca, 2009) show that an LDA based generative scheme is the best approach for expanding WordNet hypernym-hyponym structure via attribute extraction.

Ontology based extraction. (Maedche et. al, 2003) provide an ontology based information extraction technique which uses weighted finite state machines. The approach is generic to information extraction tasks and does not specialize in attribute extraction as well the finite state machines need supervision. Moreover, they use German corpora for all the evaluation. (Embley et. al, 1998) use domain based ontologies for information extraction. After choosing the relevant ontology they formulate a set of rules for extracting constants and keywords.

Tag based supervised extraction. (Ghani et al., 2006) treat each product as attribute-value pair and use a set of seed labels to induce a classification setting for extraction. Their first step is to define a set of attributes to be extracted. (Putthividhya and Hu, 2011) provide a tag based brand name extraction technique for online products. Their problem overlaps with the SEM problem as ads need brand names too. They use ebay shoes and clothing product catalog as their corpus.

Semantics and Rule based extraction Nagy and Farkash (2010) assign webpages to people based on the attributes matched among them. They manually mark relevant attributes then formulate empirical rules to extract attribute values. (Nagy and Farkas, 2008) provides logic based approach to extracting class attributes from English texts. Etzioni (2005) et. al provide an experimental study with an extraction scheme for obtaining named entities from web. Their scheme relies on domain independent extraction patterns to generate candidate named entities.

All the above approaches can be classified into two categories: a) they use a seed set for training, or b) they use a pattern or rule empirically discovered for the extraction. Due to this fact, all of the above approaches are insufficient for our requirement because they are not scalable to hitherto unseen product categories. And a prior knowledge of what the attributes are is needed in all of the approaches. Our model deals with both of these issues through utilizing the semantic clustering of words based on WordNet metrics. It is completely unsupervised in terms of seed sets or rules/patterns. The approach that comes closest to solving the SEM problem is unsupervised topic model (Blei et. al, 2003) as this too doesn’t need any seed set or assume any rules.

4. Document Collection

The datasets used for the models are product catalogs of different online product categories. This is done to make sure that the models face the same issue as in the real world SEM tasks. The real world SEM techniques use sellers’ product catalog to generate relevant ad-copies. We use 4 different online product-catalogues of 4 different sellers: 1) Furniture Catalog, 2) Clothing Catalog, 3) Watches Catalog, and 4) Beddings Catalog. We compare our model with LDA and a baseline and report the results.

The model uses four datasets, two for parameter tuning and two for testing. All Four datasets are product catalogs of

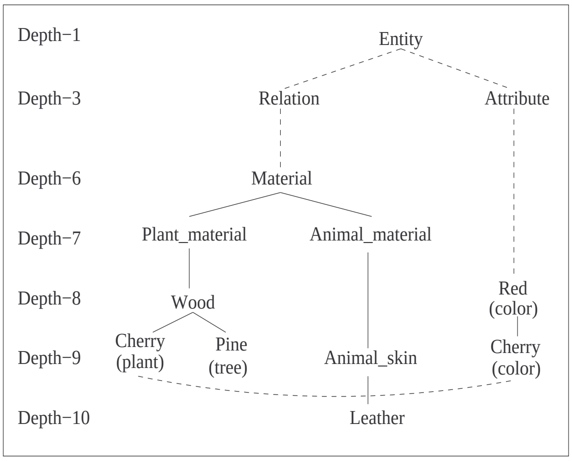


Figure 1: An example sense hierarchy for WordNet, nodes at the same height have the same depth in WordNet. The depth at each height is labeled in the left side of the figure. Immediate parents are connected with a solid line.

online products on sale. Each product entry in a catalog is one line.

Furniture Catalog. This furniture catalog has 7677 product entires and 49709 words. Each entry is a long phrasal noun eg. "Cambridge Black 25-Inch Backless Counter Swivel Stool with Black Vinyl Cushion Seat".

Clothing Catalog. This is a catalog of clothes with 16839 product entries and contains a total of 24485 words. A typical entry is: "Plus Size Full Bust".

Watches Catalog. This is a catalog of watches. It has 7982 entries and 68397 words. A typical entry looks like "Nixon Men's 'The Rocker' Stainless Steel and Leather Quartz Watch".

Beddings Catalog. This is Beddings catalog with a total of 22955 product entries and 153552 words. A typical entry here looks like "Frette Completo Letto Textured Queen Bedspread".

5. The Attribute Extraction Model

The model treats the product catalog as a bag of words. Each word w present in catalog d is assigned a probability mass $P(w|d)$ as follows:

$$P(w|d) = \frac{N_d(w)}{\sum_{\forall w \in d} N_d(w)} \tag{3}$$

where $N_d(w)$ is the count of word w in catalog d. The sense-set ψ_w for each word w is the set of all senses of w, i.e.

$$\psi_w \ = \ \{w_s : w_s \in synset(w)\} \tag{4}$$

where $synset(w)$ contains the SynSets of word w in WordNet (Miller, 1995). Each member w_s of set ψ_w is a unique sense of word w and lies in a unique SynSet of w. The catalog d is expanded to a "bag of senses", Ψ, where:

$$\Psi = \cup_{\forall w \in d} \psi_w \tag{5}$$

A naive approach to clusters these senses is to group two senses, w_{s_1} and w_{s_2} , together if $hypernym(w_{s_1}) = hypernym(w_{s_2})$, i.e. w_{s_1} and w_{s_2} are immediate siblings in WordNet hypernym tree. Figure 1 shows an example where two immediate sense siblings, "cherry" and "pine" are clustered using a common parent "wood". "wood" becomes the cluster head of this cluster. This approach can be extended to include "leather" in the cluster with "material" as the new cluster head. "material" is a valid cluster label and we propose later in this section a model that arrives at such valid cluster heads or labels.

5.1. Modeling

Aforementioned naive approach of clustering words based on WordNet sense hierarchy does not know how to arrive at valid cluster heads, i.e. when to stop adding more hierarchies to the sense tree . This task can be achieved by utilizing 2 important semantic metrics:

- **depth-metric**: the sense of a word increases in specificity as the word's depth increases in WordNet (Jiang and Conrath, 1997)

- **hop-metric**: the smaller the hop-counts between two words in the WordNet taxonomy the closer their senses are (Rada et al., 1989).

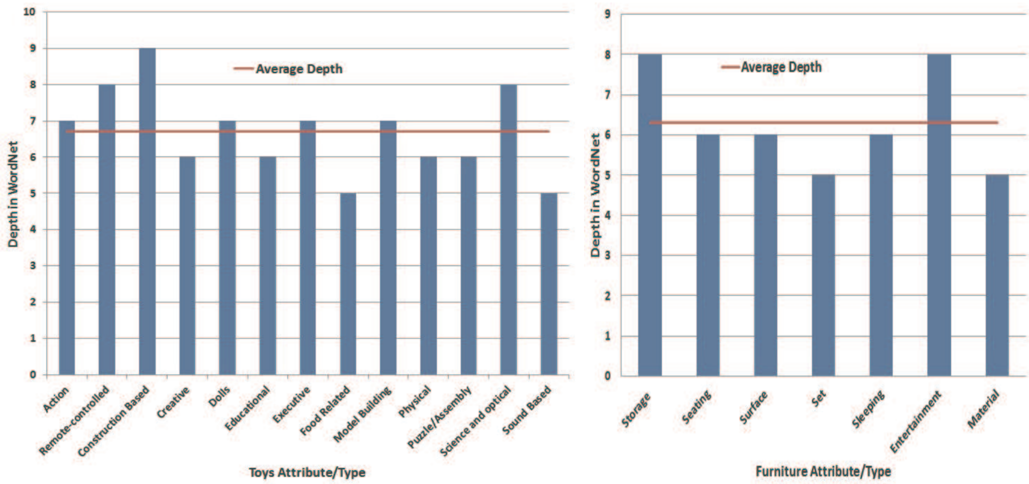
The proposed model tunes its parameters based on the above 2 metrics. The model learning has 2 phases: 1) Cluster Discovery, and 2) Cluster Pruning.

5.2. Cluster Discovery (Phase I)

Algorithm 1 describes the cluster discovery process in detail. The model iterates through each word-sense present in the "bag of senses", Ψ obtained from equation 5, and clusters them together based on WordNet's hyponym-hypernym (IS-A) relation. For each sense $s \in \Psi$, the algorithm first iterates through all the discovered clusters in cluster set Ω and checks whether $\exists C_i \in \Omega \ni s$ has a valid hypernym/hyponym relation with C_i. If $\exists C_i \in \Omega$ then s is added to C_i and $C_{i_{head}}$ is modified appropriately. If there is no such C_i then the model iterates through the hitherto unclustered senses $s_j \in \Psi$ such that s and s_j has a hypernym/hyponym relationship between them. If there exists such an s_j then a new cluster C_{new} is created with s and s_j inserted into C_{new} and $c_{new_{head}}$ appropriately initialized. This C_{new} is inserted into hitherto discovered cluster Ω. The model moves onto the next sense in Ψ and starts the above steps again. After iterating through all elements of Ψ, Ω returns with a set of candidate clusters/topics with each cluster's head assigned as label for that topic. The labels of these clusters will become our discovered attributes of the product. Each $C_i \in \Omega$ is a cluster of word-senses with a sense-hierarchy among the elements present in it.

5.3. Cluster Pruning (Phase II)

The clusters obtained in phase I contain lot of noise and are not sense specific. Table 2 shows some of the prominent clusters discovered after phase I in the Furniture Catalog. The Cluster Pruning phase deals with by parametrizing the cluster properties based on the WordNet metric defined in section 5.1.. The clusters have the following properties:



(a) Toys attributes depth

(b) Furniture attributes depth

Figure 2: WordNet depths of various attributes of Toys and Furniture catagory obtained from sample Wikipedia pages

unit	set	play	event	material	furniture	equipment	wood	colour	leather
unit	set	play	event	stuff	dresser	equipment	wood	color	leather
clark	quartet	Hole	turn	Soda	chest	Bird	Ash	Red	Kids
london	suite	set	label	Lime	bureau	glove	Hardwood	White	Buff
almond	product	Sets	carry	Products	Table	hammer	Wicker	Yellow	Suede
hamilton	Core		Case	Crib	Etagere	Set	Alder	Jade	Micro-suede
clock	Triplet		Pawn	Stool	Sleeper	Bag	Birch	Tawny	Crushed
bradley			Articulating	Salmon-colored	Seats	Wood-base	Knot	Two-Tone	Alligator
solitary			Sitting	Mocha-colored	Counter	X-base	Log	Grey	Morocco
lotus			Adornment	Mahogany-color	Buffet	Club	Driftwood	Pastel	Cordovan
prince			White-washed	Straw	Tufted-seat	Wicket	Cedar	Brown	Mocha

Table 2: The result of the model for $\kappa = 5$, before and after Pruning for Furniture category. The top 10 elements in each cluster are shown.

Cluster Depth (C_{depth}): The cluster depth is the depth of the head-node of the cluster in the WordNet sense hierarchy i.e. $C_{depth} = C_{head_{depth}}$.
Cluster Breadth ($C_{breadth}$): The cluster breadth is the vertical span of the the cluster tree in terms of WordNet depth.

$$C_{breadth} = \max(node_{depth} - C_{depth}) \tag{6}$$

where $node \in C$.
Cluster Probability Mass (C_{mass}): The probability of a sense s in catalog d is defined as, $P(s|d) = P(w|d)$ where $s \in \psi_w$ i.e. s is a sense of word w. The cluster probability mass, C_{mass} is based on this.

$$C_{mass} = \sum_{s \in C} P(s|d) \tag{7}$$

Cluster Density ($C_{density}$): The cluster density is defined as:

$$C_{density} = \frac{C_{mass}}{C_{breadth}} \tag{8}$$

Mutual Information ($MI(C_1, C_2)$): Mutual information between any two clusters C_1 and C_2 for a given catalog d measures the amount of common mass between the two clusters. A common word set, Γ_{C_1,C_2}, between C_1 and C_2 is defined as:

$$\begin{aligned}\Gamma_{C_1,C_2} &= \{w : w \in d \wedge (\exists s_1, s_2 \in \psi_w \ s.t. \\ &\quad (s_1 \in C_1 \wedge s_2 \in C_2))\}\end{aligned} \tag{9}$$

where w is a word in catalog d. The Mutual information, $MI(C_1, C_2)$ is defined as :

$$MI(C_1, C_2) = \frac{\sum_{w \in \Gamma_{C_1,C_2}} P(w|d)}{C_{1_{mass}}} \tag{10}$$

To obtain a more sense specific set of clusters with valid attribute labels, the model constraints the above defined cluster properties through 3 model parameters. These parameters are based on the semantic metrics mentioned in section 5.1.. The 3 parameters are as follows:

1. δ: This parameter regulates the depth of discovered cluster C. As observed in section 5.1., the deeper a cluster, the more sense specific it becomes. δ tunes the depth property of a cluster to get suitable product attributes as respective clusters.

2. β: In the WordNet sense-hierarchy, the sense-specificity spreads as one goes down the hierarchy. The model parameter β controls such a spread in the clusters and doesn't let it cross a threshold.

3. μ: The mutual information, $MI(C_1, C_2)$, between 2 clusters C_1 and C_2 gives an estimate of how much one cluster replicates the other. If this replication goes beyond a threshold the smaller cluster should be discarded as it does not contain any independent information of its own. The model parameter μ regulates this threshold.

ALGORITHM 1: Cluster Discovery

Initialize:
Ψ /*obtained via equation 5*/
$\Xi \leftarrow \{\}$/*stores clustered senses*/
$\Omega \leftarrow \{\}$/*set of discovered sense clusters*/
$flag \leftarrow$ false
for *each* $s \in \Psi$, **do**
  for *each* $C_i \in \Omega$, **do**
    if $hypernym(s) = C_{i_{head}}$ **then**
      insert s into cluster C_i
      $flag \leftarrow$ true
    end
    if $hypernym(C_{i_{head}}) = s$ **then**
      $C_{i_{head}} \leftarrow s$
      $flag \leftarrow$ true
    end
  end
  if $flag$ **then**
    insert s into Ξ
    remove s from Ψ
    $flag \leftarrow$ false
  end
  else
    for *each* $s_j \in \Psi$ **do**
      if $hypernym(s) = s_j$ **then**
        create new cluster C_{new}
        $C_{new_{head}} \leftarrow s_j$
        Insert s in C_{new}
        $flag \leftarrow$ true
        break
      end
      else if $hypernym(s_j) = s$ **then**
        create new cluster C_{new}
        $C_{new_{head}} \leftarrow s$
        Insert s_j in C_{new}
        $flag \leftarrow$ true
        break
      end
    end
  end
  if $flag$ **then**
    insert s, s_j into Ξ
    remove s, s_j from Ψ
    insert C_{new} into Ω
  end
end
return Ω

Figure 2 shows depths of all attributes of Toys and Furniture obtained from Wikipedia pages [1] [2]. The attributes obtained from these pages are first converted to their closest morphological noun form. The horizontal lines show the average depth of the categories which is 6.3 for Furniture and 6.7 for Toys. We can also see that the depths are also in the range of 5 and 8. This gives the intuition that the WordNet depths of product attributes are not all that random and have some patterns to them. The model captures this through the parameter δ

[1] http://en.wikipedia.org/wiki/List_of_furniture_types

[2] http://en.wikipedia.org/wiki/List_of_toys

ALGORITHM 2: Cluster Pruning

Initialize:
$\hat{\Omega} \leftarrow [\ ]$ /*array of ranked clusters*/
for *each cluster* $C \in \Omega$ **do**
  $\Lambda \leftarrow \{\ \}$
  if $C_{head_{depth}} \leq \delta \vee C_{breadth} > \beta$ **then**
    $\Lambda \leftarrow DisIntegrate(C)$
    /*function DisIntegrate defined below*/
    remove C from Ω
  end
  for *each cluster* $C_i \in \Lambda$ **do**
    insert C_i into Ω
  end
end
/* Prune for overlapping clusters*/
for *each* $C_i \in \Omega$ **do**
  for *each* $C_j \in \Omega$ **do**
    if $MI(C_i, C_j) > \mu$ **then**
      $C_{smaller} \leftarrow C_j$ **if** $C_{j_{mass}} > C_{i_{mass}}$ **then**
        $C_{smaller} \leftarrow C_i$
      end
      remove $C_{smaller}$ from Ω
    end
  end
end
/* Rank the new clusters*/
$\hat{\Omega} \leftarrow [\Omega]$
for *each* $C_i \in \hat{\Omega}$ **do**
  for *each* $C_j \in \hat{\Omega}$ **do**
    if $C_{i_{density}} \geq C_{j_{density}}$ **then**
      swap $\hat{\Omega}[i]$ and $\hat{\Omega}[j]$
    end
  end
end
return $\hat{\Omega}$
/*function DisIntegrate*/
DisIntegrate(cluster C)**:**
$\Lambda \leftarrow \{C\}$
while $\exists\, C_i \in \Lambda \wedge (C_{i_{depth}} \leq \delta \vee C_{i_{breadth}} > \beta)$ **do**
  for *(each* $node_j \in C_i \wedge (C_{i_{depth}} - node_{j_{depth}}) = 1$*)*
  do
    $C_j \leftarrow$ child cluster with head $node_j$ insert C_j into Λ
    remove C_i from Λ
  end
end
return Λ

The parameters, δ, β and μ, used in Algorithm 2 are learned over training set.

Algorithm 2 shows the steps involved in Cluster Pruning phase. Each cluster obtained form phase I is tested on the three parameters δ, β, μ defined above. A cluster which does not satisfy the constraints imposed by any of the three parameters is broken into smaller clusters, where the smaller cluster are the subtrees one hop down in the sense hierarchy in the cluster. The children of the previous cluster head are the cluster heads of the respective new clusters. When all the clusters present in cluster set, Ω, satisfy the constraints imposed by the model parameters, the model goes on to create a ranked set of clusters $\hat{\Omega}$. In the cluster set $\hat{\Omega}$, the clusters are arranged in the descending order of their cluster density defined in equation 8.

This density is modified in the cases when "hop-holes" are discovered, i.e. when the nearest child to a cluster-head is more than one hop away. In that case, the new density $C'_{density} = \frac{C_{density}}{C_{depth_{avg}} - \delta + hop\_size}$ where $C_{depth_{avg}} = \frac{\sum_{C_i \in \hat{\Omega}} C_{i_{depth}}}{|\hat{\Omega}|}$ and hop_size is the "hop-hole" size.

6. Evaluation Setup

The topics discovered are evaluated by a group of 7 independent domain experts. Each expert-labeler labels every topic discovered and assigns a "valid" or "invalid" label based on whether the topic is a valid attribute of the product. The labelers also label the words present in each valid topic as a "noisy" or "valid" member of the cluster. All the results and parameter-tuning are based on the consensus label of the experts. The consensus label for each data point is obtained via majority voting.

Evaluation Metric. We report number of valid attributes discovered η, and average cluster purity ρ_{avg} of the clusters predicted. The cluster purity of a valid cluster C, if its size is $|C|$ (eg. 100) and has v (say 90) valid words in it as defined above, is $\rho_c = \frac{v}{|C|}$ (0.9). The average cluster purity for top κ clusters is:

$$rho_{avg} = \frac{\sum_{(C \in \ valid)} \rho_c}{\kappa} \tag{11}$$

Data. The model only deals in noun senses to maintain simplicity. All words are converted to their morphologically closest noun word. Eg. "educational" is converted to "education". This does not make the model loose any original word-sense for majority of the words since the model takes all senses of a word into account. Hence all the senses of the new noun-word are taken into consideration reducing the risk of loosing an original word-sense to the minimum. The model tunes its parameters over Furniture and Clothing catalogs. It is tested over Watches and Bedding catalogs and Wikipedia Clay Toy pages.

7. Experiments

7.1. Parameter Tuning.

The parameters δ, β and μ are tuned over two online catalogs: 1) Furniture and 2) Clothing. We take top-40 clusters given by the model i.e. $\kappa = 40$ and count the number of valid clusters. The depth parameter δ is optimised without any β or μ constraints. For this best value of δ the breadth parameter β is tuned without any μ constraint. For these 2 best δ and β the optimal value of μ is tuned. Figure 3 shows the graph for the parameter tuning. The left figure shows that the $\delta = 6$ gives the most number of valid clusters for both catalogs. This result is consistent with the figure 2 where the average WordNet depths for Furniture and Toys category attributes are 6.3 and 6.7 respectively. The center figure in figure 3 shows that for the best δ (6) the optimal β lies in $[6, 8]$. Clothing doesn't show any improvement from constraint β but Furniture gains 2 more valid clusters by imposing β constraint. We take the largest β in $[6, 8]$, $\beta = 8$, as the optimal β to avoid breaking clusters unnecessarily. The right most figure in figure 3 shows the tuning graph for μ for $\delta = 6$ and $\beta = 8$. Furniture doesn't gain anything from μ but Clothing gains 5 more valid clusters by the imposition of μ. Optimal μ lies in the region $[0.6, 0.9]$. We pick $\mu = 0.7$ as optimal as that seems to be optimising for both catalog in the left most figure.

	LDA	Our Model
Bedding	0.50	0.81
Watches	0.35	0.878

Table 4: Average cluster purity ρ_{avg} for $\kappa = 10$

7.2. Test Results

The model is compared with the traditional LDA model and a baseline. The baseline is the number of valid attributes as judged by the experts in top-κ words ranked by the word count in the catalog. However, this baseline would not help the ad-copy creation problem as it just represents a possible label for an attribute set without containing any actual attributes. The word judged to be a valid attribute must be a generic enough word to be a valid label for a cluster of attributes of the product. This baseline is provided solely for comparative study. For the LDA model, each catalog d is divided randomly into N documents with each document getting $\frac{d}{N}$ catalog entries each. The results are reported for best N and optimised parameters of the LDA. Each topic obtained from LDA is a cluster of top 20 most likely words in that topic. We are looking for distinct attributes discovered thus if two topics are about the same attribute then they are counted as one valid topic.

Background. The problem of extracting the product attributes for ad-copy creation involved employees going through the catalog manually and looking for probable attributes which can be formalized in a functional way as described in equation 1. A topic model helped this manual labor by giving a probable set of topics based on the word co-occurrances in the product catalog. One would go through this probable set of topics and extract purer ones by pruning them out. For the purpose of comparison here, we do not prune the topics obtained from LDA and they are reported as "valid" or "noisy" by the experts based on the majority of words being valid or noisy.

We provide an average cluster purity (ρ_{avg}) comparison for valid clusters obtained in the top-10 ($\kappa = 10$) clusters returned by LDA and our model. The clusters reported in table 3 are for $\delta = 6$, $\beta = 8$, $\mu = 0.7$ and $\kappa = \{2, 3, 5, 10, 20, 30, 40\}$. We see that the baseline and LDA are outperformed by our model. The LDA model is only able to find "material" and "brand" attributes for watches catalog and "brand", "size" and "bedding" attributes for bedding catalog. These clusters keep occurring repeatedly in multiple topics discovered by LDA. Table 5 displays the

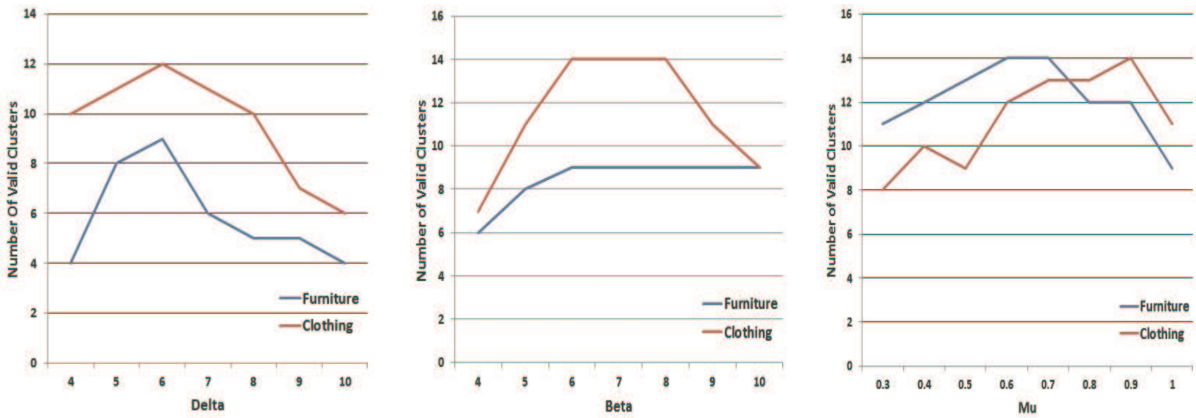


Figure 3: Tuning parameters δ(Delta), β(Beta) and μ(Mu) on Furniture and Clothing catalogs, for $\kappa = 40$.

	$\kappa = 2$	$\kappa = 3$	$\kappa = 5$	$\kappa = 10$	$\kappa = 20$	$\kappa = 30$	$\kappa = 40$
Bedding-LDA	1	1	1	2	2	3	2
Bedding-baseline	1	1	1	1	2	2	2
Bedding-Our Model	1	2	3	5	11	15	19
Watches-LDA	2	2	2	2	2	2	2
Watches-baseline	1	2	2	2	2	2	3
Watches-Our Model	2	3	4	7	8	11	14

Table 3: Number of valid clusters for Bedding and Clothing catalogs for different κ values

timepiece	metal	color	quartz	leather	jewelry	band	Material	Brand
timepiece	metal	color	quartz	leather	jewelry	band	Watch	Roamer
watch	brass	purple	rhinestone	calfskin	bead	rim	Men's	Accutron
timer	steel	red	aventurine	D-KIDS	bling	strap	Women's	Chronotech
clock	bronze	olive	topaz	Grain	pin	Rimmed	Steel	Perpetual
wristwatch	stainless	Brown	agate		band	flat	Stainless	Hush
hunter	gunmetal	salmon	Suede		bracelet	bracelet	Quartz	Crystal-accented
chronograph		blue			clip	weed	Black	Polyurethane
stopwatch		Black			chain	carabiner	Dial	Rotary
alarm		grey			gem		Strap	Luminox
chronometer		yellow			sapphire		Leather	Expansion

Table 5: Valid Clusters discovered for Watches catalog and $\kappa = 10$, the first 7 clusters are discovered by our model and the last 2 are discovered by LDA. The first row in the table is the cluster label.

valid cluster attributes discovered in top-10 clusters given by our model and LDA. We can see that these, attribute clusters are very pure in case of our model. Moreover, these attributes would be very hard to discover by a tagging or a rule based technique unless we know what we are looking for.

8. Discussion and Conclusion

We have presented here an effective mechanism for unsupervised semantics based attribute extraction. The model relies on WordNet semantics and sense-ontology and statistical and unique properties of the SEM dataset. The SEM datsets are a single catalog file containing product entries with each entry effectively a big noun phrase. A word co-occurrance based approach like LDA will not work very well here as shown earlier. The proposed model can also be used as a bootstrapping method for tag based extraction techniques. The valid attribute clusters returned by the model can be used as a seed set for the corresponding attribute set.

Though our model works very well for SEM tasks it has its limitation. It is not a generic model and will fail to extract patterns over a collection of documents. An interesting area of further exploration would be how this model performs for generic topic discovery tasks. This model can be combined with a generative scheme for topic discovery tasks such as LDA in order to make the generative process take into account the semantic properties of words. The current LDA lacks this highly desirable property.

In the present model we assigned same probability of the root word to its child senses. Another way to assign probabilities to sense would be to equally divide the original word's probability among its child sense. This scheme will also take into account the inherent ambiguity of words, i.e. words have with more senses and hence more ambiguous would pass on fewer probability mass to their each child.

Acknowledgments

We thank the anonymous reviewers for their valuable feedback. This work was supported by Texas Advanced Tech. Project (TATP) grant and Norman Hackerman Advanced Research Program (NHARP) for the first and the last author.

9. References

Freitag D. and McCallum A. 1999. *Information extraction using HMMs and shrinkage* In Proceedings of the AAAI-99 Workshop on Machine Learning for Information Extraction: 31–36.

Xiaofeng Yu, Wai Lam and Bo Chen 2009. *An Algebraic Approach to Rule-Based Information Extraction* In Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM): 325–334.

Reiss F. and Raghavan S. and Krishnamurthy R. and Huaiyu Zhu, and Vaithyanathan S. 2008. *An integrated discriminative probabilistic approach to information extraction* In Proceedings 24th International Conference on Data Engineering (ICDE): 933–942.

Eisenstein J. and Yano T. and Cohen, William W. and Smith, Noah A. Xing, Eric P.; 2011. *Structured Databases of Named Entities from Bayesian Nonparametrics* In Proceedings the EMNLP Workshop on Unsupervised Learning in NLP: 2–12.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates 2005. *Unsupervised named-entity extraction from the web: an experimental study* Artificial Intelligence Journal, 165(1): 91–134.

Fukuda K., Tamura A., Tsunoda T., and Takagi T. 1998. *Toward information extraction: identifying protein names from biological papers* Pac Symp Biocomput, 707–718.

Ellen Riloff and Rosie Jones 1999. *Learning dictionaries for information extraction by multi-level bootstrapping* Proceedings of the 16th national conference on Artificial intelligence (AAAI), 474–479.

K. Bellare, P. Talukdar, G. Kumaran, F. Pereira, M. Liberman, A. McCallum, and M. Dredze 2007. *Lightly-supervised attribute extraction* Proceedings of Machine Learning for Web Search Workshop, NIPS.

Mike Moran and Bill Hunt 2009. *Search Engine Marketing, Inc.: Driving Search Traffic to Your Company's Web Site* IBM Press, Second Edition, ISBN: 978-0-13-606868-6.

George A. Miller 1995. *WordNet: A Lexical Database for English* Communications of the ACM, 38(11): 39–41.

Jay Jiang and David Conrath 1997. *Semantic similarity based on corpus statistics and lexical taxonomy* Proceedings of International Conference on Research in Computational Linguistics, Volume 33.

Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner 1989. *Development and application of a metric on semantic nets* IEEE Transactions on Systems, Man and Cybernetics, 19(1): 17–30.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella 2003. *Kernel methods for relation extraction* The Journal of Machine Learning Research, Volume 3: 1083–1106.

Rayid Ghani, Katharina Probst, Yan Liu, Marko Krema, and Andrew Fano 2006. *Text mining for product attribute extraction* ACM SIGKDD Explorations Newsletter, 8(1).

Istvn T. Nagy and Richrd Farkas 2010. *Person attribute extraction from the textual parts of web pages* Third Web People Search Evaluation Forum (WePS-3), CLEF.

Benjamin Van Durme, Ting Qian, and Lenhart Schubert 2008. *Class-driven attribute extraction* In Proceedings of the 22nd International Conference on Computational Linguistics (COLING): 921–928.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan 2003. *Latent dirichlet allocation* The Journal of Machine Learning Research, Volume 8: 993–1022.

Duangmanee Putthividhya and Junling Hu 2011. *Bootstrapped Named Entity Recognition for Product Attribute Extraction* Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing: 1557–1567.

Joseph Reisinger and Marius Pasca 2009. *Latent Varibale Model for Concept Attribute Attachment* In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: 620–628.

Alexander Maedche, Gnter Neumann, and Steffen Staab 2003. *Bootstrapping an ontology-based information extraction system* Intelligent exploration of the web Physica-Verlag GmbH, Heidelberg, Germany, ISBN:3-7908-1529-2.

David W. Embley, Douglas M. Campbell, Randy D. Smith, and Stephen W. Liddle 1998. *Ontology-based extraction and structuring of information from data-rich unstructured documents* In Proceedings of the 7th international Conference on Information and Knowledge Management (CIKM): 52–59.