Entity-balanced Gaussian pLSA for Automated Comparison

Danish Contractor*
IIT Delhi & IBM Research
New Delhi, India
dcontrac@in.ibm.com

Mausam and **Parag Singla**
IIT Delhi
New Delhi, India
{mausam,parags}@cse.iitd.ac.in

Abstract

Community created content (e.g., product descriptions, reviews) typically discusses one entity at a time and it can be hard as well as time consuming for a user to compare two or more entities. In response, we define a novel task of automatically generating *entity comparisons* from text. Our output is a table that semantically clusters descriptive phrases about entities. Our clustering algorithm is a Gaussian extension of probabilistic latent semantic analysis (pLSA), in which each phrase is represented in word vector embedding space. In addition, our algorithm attempts to balance information about entities in each cluster to generate meaningful comparison tables, where possible. We test our system's effectiveness on two domains, travel articles and movie reviews, and find that entity-balanced clusters are strongly preferred by users.

1 Introduction

The proliferation of Web 2.0 has enabled ready access to large amounts of community created content, such as status messages, blogs, wikis, and reviews. These form an important source of knowledge in our day to day decision making, such as deciding which restaurant to try, or which movie to watch, or which city to visit etc. Unfortunately, such content typically focuses on one real world entity at a time, whereas, a user deciding between alternatives is most interested in a *comparative* analysis of strengths and weaknesses of each.

There have been some recent attempts to create comparisons using expert knowledge, but generating such comparisons manually does not scale – even pairwise comparisons are quadratic in the number of entities. Few automated comparisons for specific products with pre-defined attributes (e.g., laptops, cameras) exist; they are typically powered by existing structured knowledge bases. To the best of our knowledge, prior work on automatically generating comparisons for arbitrary domains from unstructured text, does not exist.

We define a novel task of generating *entity comparisons* from textual corpora in which each document describes one entity at a time. For broad applicability, we do not restrict ourselves to a pre-defined ontology; instead, we use textual phrases that describe entities as our unit of information. We call these *descriptive phrases* – they encompass general attribute-value phrases, opinion phrases, and other descriptions of the facets of an entity. We generate entity comparisons in a tabular form where the phrases are organized semantically, thus, allowing for direct comparisons. Figure 1 shows a sample city comparison generated by our system for tourism.

Our comparison generation algorithm extracts descriptive phrases per entity and clusters them into semantic groups. We perform clustering via a topic model, where phrases from an entity are combined into one document. The topics identify prominent facets of the entities. Unfortunately, since the number of entities being compared is usually small, just statistical co-occurrence of words and phrases is not sufficient to identify good topics. In response, we use vector embeddings of descriptive phrases and

[*] This work was carried out as part of PhD research at IIT Delhi. The author is also a regular employee at IBM Research.

Proceedings of NAACL-HLT 2016, pages 69–79,
San Diego, California, June 12-17, 2016. ©2016 Association for Computational Linguistics

employ a *Gaussian extension* of probabilistic latent semantic analysis (pLSA) over these vectors.

We also modify Gaussian pLSA to additionally incorporate an *entity-balance* term, preferring topics in which phrases from the entities are represented in a proportionate measure. The balance term trades off the discovery of unique facets for each entity with that of common facets. This enables direct comparison between entities leading to an overall improved comparison table. Since the balance term is only a preference (not a constraint), it still allows the algorithm to exhibit clusters which may be sparsely represented (or not represented at all) in one of the entities.

We demonstrate the usefulness of our ideas on two domains – tourism and movies. Based on user experiments, we find that the entity-balanced model outputs much better comparisons as compared to an entity-oblivious model such as GMM. In summary, our paper makes the following contributions:

- We define a novel task of generating entity comparisons from a corpus that describes entities individually.
- We present the first system to output such a comparison. Our system runs Gaussian pLSA over the vector embeddings of extracted phrases, and preferentially tries to balance the entities in each topic.
- Human subject evaluations using Amazon Mechanical Turk (AMT) demonstrate that AMT workers overwhelmingly prefer comparisons generated using entity-balanced Gaussian pLSA compared to entity-oblivious clustering.

2 Related Work

Recently, the internet has seen a growth in websites offering comparisons for different entities. Product websites such as eBay maintain comparisons for products. Google also outputs pre-built comparisons between common entities when queried with the word "vs." between them. Both of these output purely structured attribute-value information and are unable to compare along more qualitative and descriptive dimensions such as ease of living or quality of nightlife when comparing cities, for example. Other websites such as WikiVS[1] contain user-contributed comparisons that have been categorized based on the nature of the entities being compared. These are manually curated and therefore do not scale to the quadratic number of entity pairs.

Cluster Labels	**Granada (Spain)**	**New York City (U.S.)**
art, arch.	moorish architecture religious art fine art beautiful architecture	contemporary art modern american art medieval art egyptian art
palace, courtyard	brick-walled courtyard lovely courtyard area nasrid royal palace alhambra palace	
museum, finest	alhambra museum archaeological museum world heritage site splendid arabic shops	fine art museums guggenheim museum islamic art collection metropolitan museum
gardens, park	partal gardens palace gardens pleasant gardens moorish style gardens	flushing meadows park central park renowned gardens natl. recreational area

Figure 1: Sample comparison (abridged) between Granada and New York generated by our system. A quick look reveals that that both cities have a nice set of museums and gardens to visit, while palaces and courtyards are only in Granada. Granada's art and architecture are more ornamental, whereas New York's might be more contemporary.

Perhaps the most closely related work to ours is the field of contrastive opinion mining and summarization (Kim et al., 2011; Liu and Zhang, 2012). Examples include extraction of contrastive sentiments on a product (Lerman and McDonald, 2009) and summarization of opinionated political articles (Paul et al., 2010). Contrastive opinion mining extracts contrasting view points about a *single* entity or event instead of comparing multiple ones. A recent preliminary study extends this for comparing reviews of two products (Sipos and Joachims, 2013). It uses a supervised method for learning sentence alignments per product-type, and does not organize various opinions for an entity via clustering.

Other related work includes comparative text mining tasks where document collections are analyzed to extract shared topics or themes (Zhai et al., 2004). Since such methods only identify latent topics for the full document collection, they can't be directly used for a specific comparison task.

Since our system is a combination of IE and clustering, we briefly describe related approaches for these subtasks.

Information Extraction: Our work is related to

[1] http://www.wikivs.com/wiki/Main_Page

the vast literature in information extraction, in particular Open IE (Banko et al., 2007). Our use of POS patterns for extracting domain-specific descriptive phrases is similar in spirit to ReVerb's patterns for relation extraction (Etzioni et al., 2011) and adjective-noun bigrams for fine grained attribute extraction (Huang et al., 2012; Yatani et al., 2011). Adapting the literature on entity set expansion (Pantel et al., 2009; Voorhees, 1994; Natsev et al., 2007), our system expands seed nouns for broader coverage. We use Wordnet and distributional similarity-based approaches for this (Curran, 2003; Voorhees, 1994).

Clustering: Our entity-balanced clustering algorithm is related but different from previous work on *balanced* clustering. Prior work (Banerjee and Ghosh, 2006; Yuepeng et al., 2011) has focused on generating different clusters to be equi-sized. Other work (Zhu et al., 2010; Ganganath et al., 2014) enforces size constraints on clusters. Our idea of balance, on the other hand, is targeted towards a better comparison and prefers that entities are well represented (balanced) in each cluster.

3 Task & System Description

Our motivation is to concisely compare two or more entities to aid a user's decision making. We make several choices in our task definition to help with this goal. First, we decide to output comparisons using a succinct tabular representation (see Figure 1). It has higher information density compared to, say, writing a natural language comparison summary.

Second, our unit of information is a *descriptive phrase*. We define it as any short phrase that describes an entity – these include attribute-value pairs (e.g., "Greek art"), opinion phrases (e.g., "spectacular views"), as well as other descriptions (e.g., "oldest church of Europe").

Third, for better readability, our table must organize the information coherently along various aspects relevant for a comparison. We achieve this by grouping related descriptive phrases. The choice of aspects should be dependent on the specific entities being compared, e.g., the facet of "beaches" may split into "water activities" and "beach types" for Jamaica v.s. Hawaii, but not for San Francisco v.s. Bombay.

Moreover, comparisons are meant to highlight both the similarities and the differences between entities. We therefore need to trade-off the discovery of unique facets of an entity with those which are common to the entities being compared. Thus, while clusters that balance the entities are preferable, it is also acceptable to have clusters where one of the entities is sparsely represented (or not represented at all). This would happen in situations where that entity does not express a particular aspect and other entities do. Comparisons must trade off semantic coherence of facets with entity-balance in each facet.

Last, but not the least, since the comparisons are targeted to aiding user's decision making, understanding her intent is important. As an example, the user may be interested in city-comparison for the purpose of tourism, or for choosing a city to live in. Descriptive phrases for the former could be related to sightseeing, shopping, etc., but for the latter they may cover aspects such as living expenses, transportation, and pollution. We accommodate this necessity by allowing minimal human supervision for specifying user intent. This supervision can come in forms such as an intent-relevant seed noun list, or topic-level annotation following unsupervised topic modeling, etc. This supervision further guides descriptive phrase extraction.

System Architecture: Our system consists of a pipeline of information extraction, clustering, cluster labeling and phrase ordering. IE extracts descriptive phrases relevant to user-intent and we develop a new clustering algorithm that produces better comparisons by balancing the entities in each cluster. We identify cluster labels based on the most frequent words in a cluster. We order phrases within a cluster based on the distance from the centroid. We now describe our IE and clustering techniques in detail.

3.1 Information Extraction

Our IE pipeline works in two steps. We first extract descriptive phrases via POS patterns and then filter out the non-topical phrases. For filtering, first we create a seed list of relevant nouns via minimal human supervision, which are then expanded by itemset expansion. Descriptive phrases with a noun in the expanded list are retained, and rest are filtered.

Preliminary analysis on a devset revealed that a large fraction of descriptive phrases are noun phrases (NPs). We first extract all NP chunks from the collection and, additionally, using POS tags, extract any adjective-noun bigrams that are part of a bigger NP chunk, or missed due to chunking errors. This forms the initial set of descriptive phrases.

Filtering for User Intent:

These descriptive phrases include those that are not relevant for user intent such as "excellent schools" for tourism. We filter these phrases by matching them to a list of intent-specific nouns. This list is created by first curating a seed list and then expanding it using item-set expansion. We employ two methods to obtain a seed list for specifying user intent: (1) a list of user-specified seed nouns, and (2) a labeling of LDA topics based on top words in each topic.

In the first approach we get the seed nouns directly from the domain expert. Our system supports the process by identifying frequent nouns and showing those to the annotator to annotate. For our tourism system, an author spent about three hours to produce a list of 100 seed nouns.

Since this process requires significant effort per user intent, we also investigate a semi-automatic approach in which we run Latent Dirichlet Allocation (LDA) (Blei et al., 2003) on the whole phrase list. We then show the top 20 words in each topic and ask the annotator to provide only topic-level annotations. We treat the top 15 words from each positively labeled topic to be in the seed set. Since the number of topics is usually not that large, this significantly reduces the time required for annotation. E.g., we ran LDA with 20 topics and it took about 10 mins. for an author to annotate them. However, the seed nouns are noisier due to noise in LDA.

Seed List Expansion: Finally, we use ideas from item-set expansion to expand the seed list for improved coverage. We implement two approaches for this step. In the first method (WN) we use Wordnet (Miller, 1995) to include words that are a direct hop away from the seed nouns. In the second approach (WV), we use word-vector embeddings (Collobert et al., 2011) and include top 10 neighbors of each seed in our expanded list. The expansions capture near-synonyms and topically related words.

IE Experiments: We now present comparisons of various IE methods on a small development set. We selected seven WikiTravel[2] articles (each article is on one city) and manually annotated an exhaustive set of descriptive phrases. This forms our devset for IE comparisons.

Method	Prec.	Recall	F1
All nouns	0.53	**0.67**	**0.59**
Seed Nouns only (Manual)	**0.77**	0.32	0.44
Seed (Manual) + WN	0.71	0.35	0.46
Seed (Manual) + WV	0.70	0.40	0.49
Seed Nouns only (LDA)	0.74	0.19	0.30
Seed (LDA) + WN	0.74	0.20	0.31
Seed (LDA) + WV	0.74	0.26	0.38

Table 1: Quality of extracted descriptive phrases on a devset

We chose various parameters in our IE systems so that our precision never drops below 0.70. For example, we used K=15 for choosing the top words from LDA into seed list. We use this target precision, because we believe that for any human-facing system the precision needs to be high for it to be considered acceptable by people.

Table 1 compares the performance of the various IE methods. Not surprisingly, we find that manual seed lists obtain a much higher recall as compared to LDA seeds, at approximately the same level of precision. Both Wordnet and word-vector improve the recall substantially, though vectors are more effective. The recall of all nouns is only 0.67 because a large number of descriptive phrases were larger n-grams (not just adjective-noun bigrams) and were missed due to chunking errors.

3.2 Building Clusters for Comparison

Our next task is to construct meaningful comparisons using these phrases. A useful comparison of entities should organize the available information in a way that is easy to comprehend by the user. Towards this goal, we group the related descriptive phrases across a number of clusters. But simply having a good clustering of descriptive phrases may not be enough. We would like to have a clustering that explicitly captures the individual characteristics of each of the entities as well as makes the relative strengths and weaknesses of each entity apparent. For example, Figure 2 (Right) shows three different clusterings of phrases from two cities; phrases from each city are in a different color. Here, the third clus-

[2] www.wikitravel.com

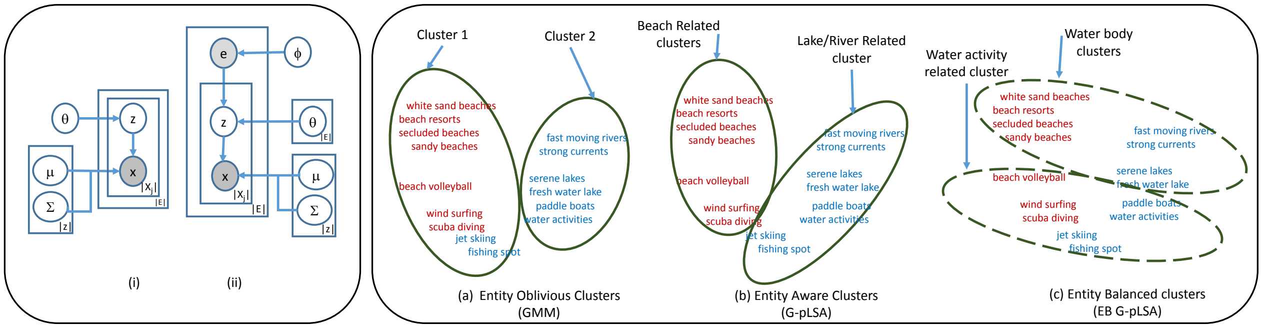

Figure 2: Left: Plate Notation of (i) Standard Gaussian Mixture Model (ii) Gaussian pLSA (and entity balanced Gaussian pLSA) **Right**: Three alternative clusterings (a), (b), (c) for descriptive phrases from two cities – each color is a different city. We prefer clusters shown in (c) as they balance information from both entities

tering is most appropriate for comparison, because not only is it a good clustering of descriptive phrases from each city considered separately, but the clusters produced also have *entity-balance*, i.e., the clusters produced have a good *balance* of both cities; both of these are key elements of comparison.

We first observe that a topic model such as Probabilistic Latent Semantic Analysis (pLSA) is a good fit to our clustering problem. In pLSA documents are characterized as mixtures of topics and topics as distributions over words. For our problem, we could combine all phrases for an entity into one document, and run pLSA to identify a coherent set of topics, which can then be used as clusters. Such a model will allow different entities to express topics in different proportions.

We note that LDA, which is a strict generalization of pLSA[3], is, in general, not a good fit for our task. LDA typically uses a *sparse* Dirichlet prior on document-topic distribution, which would not be appropriate since for comparison we would like to represent each entity in as many topics as possible.

Unfortunately, a direct application of pLSA may not yield good results. This is because typically the number of entities being compared (i.e., the number of documents in pLSA) is very small (often 2), therefore, there isn't enough statistical regularity to find good coherent topics. The alternative proposition of learning topics on the whole corpus isn't very appealing either, since that will learn global topics and not the topics particularly meaningful for the current comparison at hand.

In response, we exploit the availability of pretrained word vectors as a source of background semantic knowledge for every phrase, and generalize the pLSA model to *Gaussian pLSA* (G-pLSA). We construct a vector representation for each descriptive phrase by averaging the word-vectors of individual words in a phrase (Mikolov et al., 2013)[4]. Thus, this model is pLSA with each topic-word distribution represented as a Gaussian distribution over descriptive phrases in the embedding space. This model is also similar to the recently introduced Gaussian LDA model (Das et al., 2015), but without LDA's Dirichlet priors as discussed above.

Gaussian pLSA has several advantages for our task. First, it can meaningfully learn topics only for the entities being compared, instead of needing to learn a global topic model over the whole corpus. Second, due to additional context from word vectors, the topics are expected to be much more coherent compared to traditional topic models for cases when the underlying corpus is small, as in our case. Finally, in our model the vectors are generated from a Gaussian distribution and that helps capture the *theme* of the cluster directly by enabling a centroid computation in the embedding space. This is especially useful for identifying and ranking important descriptive phrases per cluster while generating the comparison table.

Let $x_j^{(i)}, z_j^{(i)}$ denote the values of the i^{th} phrase and the corresponding cluster (topic) id, respectively, for the j^{th} entity e_j. Then, the log-likelihood

[3] LDA with uniform Dirichlet prior is equivalent to pLSA

[4] We use the pre-trained 300 dimension vectors available at http://code.google.com/p/word2vec/

$L(\Theta)$ of the observed data can be written as:

$$\sum_{j=1}^{|E|}\sum_{i=1}^{|X_j|}\log\Bigg[\sum_{z_j^{(i)}=1}^{|Z|} P(x_j^{(i)}|z_j^{(i)};\Theta)\cdot P(z_j^{(i)}|e_j;\Theta)\cdot P(e_j;\Theta)\Bigg] \tag{1}$$

Here, $|X_j|$ and $|Z|$ are the total number of phrases and clusters[5] respectively, for a given entity e_j and, $|E|$ is the total number of entities being considered for comparison. Θ denotes the vector of all the parameters. We optimize the expression $L(\Theta)$ using EM and estimate the parameters of the model. As can be seen, the clusters are shared across entities, and the phrases generated are independent of the entity given, a cluster and the entities themselves are free to exhibit clusters in different proportions.

We also note just as pLSA can be seen as a natural extension of mixture of unigrams (Blei et al., 2003), Gaussian pLSA is an extension from the Gaussian Mixture Model (GMM) which is entity-oblivious. GMM generates each phrase independent of the entity it came from and hence, distributes entity phrases arbitrarily across clusters. We use GMM as a baseline for our experiments. Figure 2 (left) illustrates the two models in plate notation.

Entity-Balanced Gaussian pLSA: Vanilla Gaussian pLSA may not always lead to a good clustering for comparison since the expression above does not involve any term to balance the entity-information in clusters, as motivated earlier. Thus, we incorporate a regularizer term to have a good balance (proportion) of entities in each cluster (see Figure 2 (right) (c)) resulting in our final model for comparison called *Entity-Balanced Gaussian pLSA (EB G-pLSA)*. The plate notation for EB G-pLSA is identical to G-pLSA.

Our regularizer is a function of the KL-divergence between multinomial distributions for every pair of entities. KL-divergence $KL(P||Q)$ between two discrete distributions $P(x)$ and $Q(x)$ is defined as $\sum_l P(x_l)log\left(\frac{P(x_l)}{Q(x_l)}\right)$. Its an asymmetric measure of similarity and is equal to 0 when the two distributions are identical (and greater than 0 otherwise). Symmetric KL-divergence is defined as $Sym\text{-}KL(P,Q) = KL(P||Q) + KL(Q||P)$.

Let $P_{\theta_j}(z|e_j)$ and $P_{\theta_k}(z|e_k)$ denote the multinomial distributions for generating the cluster id z given the entities e_j and e_k, respectively. Here, θ_j and θ_k denote the respective multinomial parameters. We add a regularizer term to the log-likelihood minimizing the sum of symmetric KL-divergence between the distributions $P_{\theta_j}(z|e_j)$ and $P_{\theta_k}(z|e_k)$ for every pair of entities e_j and e_k. Adding this regularizer requires the multinomial distributions to be similar to each other, thereby preferring balanced clusters over unbalanced ones. Our regularized average log-likelihood can be written as:

$$L_{reg}^{avg}(\Theta) = \frac{1}{|M|}L(\Theta) - \alpha\cdot\left[\sum_{j,k=1|j<k}^{|E|} Sym\text{-}KL(P_{\theta_j},P_{\theta_k})\right] \tag{2}$$

$L(\Theta)$ is the total log-likelihood as defined in the previous equation. $M = \sum_{j=1}^{|E|}|X_j|$ and $|E|$ is the total number of entities being compared. α is a constant controlling the weight of the regularizer. Note that we add the regularizer term to the *average* log-likelihood (instead of the total log-likelihood) in order to have the same regularizer value for comparisons having varying number of data points (descriptive phrases). This is important to obtain a single value of α which would work well across different entity comparisons. In our experiments, α was tuned using held-out data and was found to be robust to small perturbations.

We use standard EM to optimize the regularized log-likelihood. Since the regularizer does not have any hidden variables, E-step is identical to the one for the unregularized case. During M-step, the values maximizing the mean parameters μ_z and the ϕ parameter can be obtained analytically. There is no closed form solution for the parameters θ_j, θ_k. We perform gradient descent to optimize these parameters during the M-step. In our experiments, we did not estimate the co-variance matrices Σ_z and kept them fixed as a diagonal matrix with the diagonal entry (variance) being 0.1. We did not learn the co-variance matrices as that would have increased the number of parameters substantially, and thus, had the danger of over fitting. The small value of the variance chosen was to ensure less overlap between different clusters.

Clustering Experiments We conducted preliminary experiments to compare the performance of GMM (vanilla Gaussian mixture modeling using word vec-

[5] Note that number of clusters for all entities will be the same i.e $|Z_j| = Z$ for all j

	GMM	G-pLSA	EB G-pLSA
f-measure	0.42	0.43	0.44
pairwise accuracy	0.66	0.65	0.76

Table 2: Comparing clustering methods on development set

tors) with G-pLSA and EB G-pLSA on a development set consisting of 5 random city pairs. The descriptive phrases were constructed using the automated seed list as described in IE Section. We manually created the gold standard clusterings. The number of clusters was set to the number in the gold set for each of the city pairs.

We used f-measure and pairwise accuracy to evaluate the deviation from the gold standard for the clusterings produced by each of the algorithms. Table 2 shows the results. EB G-pLSA performs better than the other two algorithms on both the metrics, and especially on pairwise accuracy. Performance of G-pLSA is very similar to GMM.

4 Human Subject Evaluations

In order to evaluate the usefulness of our system we conducted extensive experiments on Amazon Mechanical Turk (AMT). Our experiments answer the following questions. (1) Are comparisons generated using our clustering methods G-pLSA and EB G-pLSA preferred by users against the entity oblivious baseline of GMM? (2) Are our system-generated comparison tables helpful to people for the task of entity comparison?

Datasets & System Settings: We experiment[6] on two datasets – tourism and movies. For tourism, we downloaded a collection of 16,785 travel articles from WikiTravel. The website contains articles that have been collaboratively written by Web users. Each article describes a city or a larger geographic area that is of interest to tourists. In addition, all articles contain sections[7] describing different aspects of a city from a tourism point of view (e.g., places to see, transportation, shopping and eating). For our proof of concept, we performed IE only on the 'places to see' sections.

For Movies dataset, we used the Amazon review data set (Leskovec and Krevl, 2014). It has over 7.9 million reviews for 250,000 movies. We combined all the reviews for a movie, thus, generating a large review document per movie. This dataset is much noisier compared to WikiTravel due to presence of slang, incorrect grammar, sarcasm, etc. In addition, users also tend to compare and contrast while reviewing movies so there are even references to other movies. As a result, the descriptive phrases extracted were much more noisy.

For the time consuming manual seed list setting of our IE system, we only use the tourism dataset. For movies, we generate seeds using annotation over LDA topics only. For all systems we use word-vectors to expand the seed list.

For each table, we generated k clusters where k was determined using a heuristic[8] (Mardia et al., 1980), and we displayed at most 30 phrases per cluster. We did not display any cluster that had less than 4 phrases.

4.1 Evaluation of Clustering Algorithms

In order to examine whether clustering using EB G-pLSA indeed produces best comparison tables, we conducted a human evaluation task on Amazon Mechanical Turk (AMT) where users of our system were asked to indicate their preference between two comparison tables. Since we have three systems we performed this pairwise study thrice. In each study, two comparison tables were generated from different systems. For each entity-pair we asked four workers each to select which comparison table they preferred. The order of the tables was randomized to remove any biasing effect. We paid $0.3 for each table comparison. Table 3 reports the results for both domains where descriptive phrases were generated using LDA+WV.

On 30 city-pairs in the Tourism domain, workers preferred the comparison tables generated using EB G-pLSA 53% of the time and GMM was preferred only 13% (the rest were ties). It is worthwhile to note that whereas in 20% of the comparisons, EB G-pLSA had a clear 4-0 margin, there was no such comparison where all the workers preferred the GMM model. We also requested users to provide the reasons for their preferences. While most users specified a non-informative reason such as "like it better", some users gave specific reasons such as "subdivides the parts I find useful into more specific

[6] Code and data available on request

[7] http://wikitravel.org/en/Wikitravel:Article_templates/Sections

[8] No. of clusters = square root of half the number of phrases

Domain	Total pairs	EB G-pLSA Win		GMM Win		EB G-pLSA Win		G-pLSA Win		G-pLSA Win		GMM Win	
		4-0	3-1	1-3	0-4	4-0	3-1	1-3	0-4	4-0	3-1	1-3	0-4
Tourism	30	20%	33%	13%	0%	13%	30%	30%	0%	17%	27%	13%	3%
Movies	20	20%	35%	10%	0%	5%	35%	15%	5%	5%	45%	15%	5%

Table 3: User preference win-loss statistics for different clustering methods on both city and movie comparison task using the same IE system. Both EB G-pLSA and G-pLSA significantly outperform the baseline GMM model. EB G-pLSA has some edge over the G-pLSA model. Note: Ties have not been shown in the table.

categories" and "easy to understand and more specific points of comparison". Our results also show that G-pLSA is a distinct improvement over GMM (44% vs. 16%). EB G-pLSA had a marginal edge over G-pLSA (43% vs. 30%).

On movies domain, we report results on 20 movie-pairs and we again found an overwhelming preference for the system using EB G-pLSA for clustering. 55% of the time, the output of EB G-pLSA was preferred over GMM's 10%. Other comparisons between G-pLSA and GMM, and between our G-pLSA and EB G-pLSA systems also follow trends similar to tourism domain. The performance of EB G-pLSA is statistically significantly better than GMM for both the tourism and the movie datasets, with p values being less than 0.00004 and 0.002, respectively, using a one-sided students t-test. This strong preference suggests that the clustering induced by incorporating entity balance in the clusters produces much better comparison tables.

4.2 Value of Comparison Tables

The goal of our experiments in this section was to assess whether our comparison tables add value to some realistic task and to understand the overall usefulness of our system. To our knowledge there are no other automated systems comparing cities for tourism (or movies), hence we could not evaluate our system against existing approaches. Therefore, we decided to evaluate the benefit of the output generated by our system (i.e., comparison tables) against reading the original WikiTravel articles. For fairness we only use the 'places to see' sections from WikiTravel, since that was the raw text used in generating comparison tables in the first place.

Since the comparisons are generated automatically, people may not find them understandable, or there may be missing valuable information. We test this in a human subject evaluation. We adapt the evaluation methodology developed recently for contrasting multiple ways of presenting information and testing the overall learning of the subjects (Shahaf et al., 2012; Christensen et al., 2014). The evaluation is divided into two parts. In the first part the workers are given a limited time to read the information provided (articles or comparison tables) for an entity-pair. They are then asked to write a short 150-300 word summary contrasting different aspects of the two entities. Each user writes two summaries, one based on articles and the other based on our table. Our study pairs two users such that if user1 read the articles for city pair 1 and the table for city pair 2, their partner user will see the reverse. The workers were additionally asked which knowledge source they preferred and why.

Making a worker create summaries using both information sources helps reduce the effect of worker comprehension and skill in the evaluation of our task, as each worker contributes to summaries created using our system as well as the baseline. In order to reduce the effects of any sequence bias, half the mechanical turk workers were first shown the output of our system followed by the articles and the other half (partners) were shown content the other way around.

In the second part of this experiment we directly compare the knowledge acquisition of these workers. In particular, we ask a different set of workers to evaluate the summaries created by the partnered workers. In each task, a worker has to compare two summaries for the same entity-pair, one created using tables by one worker and other created using articles by their partner. Each summary pair was shown to four different users and each of them was asked to select the summary they preferred for comparing and contrasting the entities. Since we perform this experiment on Tourism data, the MTurk task descriptions explained that the intent of the comparison is tourism and their summaries or preferences must be from that perspective.

4.2.1 Results

We performed this evaluation on twenty city pairs using both our information extraction methods i.e. Manual+WV expansion (referred as TABLE-M) and LDA+WV expansion (referred as TABLE-LDA) along with the EB G-pLSA method for clustering. The city pairs were chosen such that the cities are related but not too similar, and the workers would likely not have thought of the specific comparisons before.

We found that in the first part where workers were given 10 minutes to create the summaries, they on average asked for 30% more time to create the summaries when information was presented as article. This supports our belief that our system-generated tables successfully reduce information overload. It also suggests that the structure added by the system (clusters) was useful for the comparison task and reduced workers' cognitive load.

We now present the results for the second part of the study in which workers evaluated the comparison summaries written by the workers in the first part. Within 20 city-pairs, summaries for 5 city pairs (25%) generated based on TABLE-M were preferred and 5 (25%) generated based on original articles were chosen. The workers were indifferent in 10 of the city pairs (both summaries got two votes each). This shows that despite having a very high compression ratio, workers still managed to create summaries that were comparable in quality to those created by reading original documents. We repeated the same study using TABLE-LDA and found that summaries for 8 city pairs (40%) generated based on TABLE-LDA were preferred and 5 (25%) generated based on original articles were chosen. The workers were indifferent in 6 of the city pairs (both summaries got two votes each).

We did not repeat this experiment using the Movies data set as the source articles were concatenated reviews with no structure and it would not be surprising that users prefer our system. In summary, we find that both our systems convey adequate and useful information in the comparisons and the summaries generated by users using our systems were found to be as good as the ones created by users reading the full articles.

5 Conclusions

We define a novel task of automatically generating tabular entity comparisons from unstructured text. We also implement the first system for this task that first extracts descriptive phrases from text and then clusters them to generate comparison tables. Our clustering algorithm is a Gaussian extension of p-LSA, where the descriptive phrases are represented using embeddings in the word vector space. In order to have a better comparison between entities, we incorporate a balance term which prefers clusters where entities are proportionately represented.

We perform extensive human-subject evaluations for our systems over Amazon Mechanical Turk (AMT) on two datasets – tourism and movies. We find that AMT workers overwhelmingly prefer EB G-pLSA based comparisons over GMM-based. We also assess the value of our generated comparisons over reading the original articles. We find that while both sets of workers learned as much, the workers viewing tables asked for less additional time to narrate a comparison in words. Overall, we believe that comparison tables add value for users deciding between multiple entities. In the future we wish to perform joint extraction and clustering instead of our current pipelined approach.

Acknowledgments

We would like to thank the users of our system: Bhadra Mani, Dinesh Khandelwal, Eshita Sharma, Kuntal Dey, Leela Muthana, Noira Khan, Samuel Kumar, Seher Contractor and Prachi Jain, and the anonymous Amazon mechnical turk workers for their evaluation and insights. We would also like to thank Ankit Anand, Happy Mittal and anonymous reviewers for their suggestions on improving the paper. The work was supported by IBM Research, Google language understanding and knowledge discovery focused research grants to Mausam, a KISTI grant and a Bloomberg grant also to Mausam. We would also like to acknowledge the IBM Research India PhD program that enables the first author to pursue the PhD at IIT Delhi.

References

Arindam Banerjee and Joydeep Ghosh. 2006. Scalable clustering algorithms with balancing constraints. *Data Min. Knowl. Discov.*, 13(3):365–395, November.

Michele Banko, Michael J Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *IN IJCAI*, pages 2670–2676.

David M. Blei, Andrew Y. Ng, Michael I. Jordan, and John Lafferty. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:2003.

Janara Christensen, Stephen Soderland, Gagan Bansal, and Mausam. 2014. Hierarchical summarization: Scaling up multi-document summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 902–912.

R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

James Richard Curran. 2003. *From Distributional to Semantic Similarity*. Ph.D. thesis, Institute for Communicating and Collaborative Systems School of Informatics University of Edinburgh.

Rajarshi Das, Manzil Zaheer, and Chris Dyer. 2015. Gaussian lda for topic models with word embeddings. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 795–804, Beijing, China, July. Association for Computational Linguistics.

Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam. 2011. Open Information Extraction: the Second Generation. In *International Joint Conference on Artificial Intelligence (IJCAI)*, Barcelona, Spain, July.

N. Ganganath, Chi-Tsun Cheng, and C.K. Tse. 2014. Data clustering with cluster size constraints using a modified k-means algorithm. In *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2014 International Conference on*, pages 158–161, Oct.

Jeff Huang, Oren Etzioni, Luke Zettlemoyer, Kevin Clark, and Christian Lee. 2012. Revminer: An extractive interface for navigating reviews on a smartphone. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, pages 3–12, New York, NY, USA. ACM.

Hyun Duk Kim, Kavita Ganesan, Parikshit Sondhi, and ChengXiang Zhai. 2011. Comprehensive review of opinion summarization.

Kevin Lerman and Ryan McDonald. 2009. Contrastive summarization: An experiment with consumer reviews. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, NAACL-Short '09, pages 113–116, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jure Leskovec and Andrej Krevl. 2014. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`, June.

Bing Liu and Lei Zhang. 2012. A survey of opinion mining and sentiment analysis. pages 415–463.

K. V. Mardia, J. T. Kent, and J. M. Bibby. 1980. *Multivariate Analysis (Probability and Mathematical Statistics)*. Academic Press, 1 edition.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November.

Apostol (Paul) Natsev, Alexander Haubold, Jelena Tešić, Lexing Xie, and Rong Yan. 2007. Semantic concept-based query expansion and re-ranking for multimedia retrieval. In *Proceedings of the 15th International Conference on Multimedia*, MULTIMEDIA '07, pages 991–1000, New York, NY, USA. ACM.

Patrick Pantel, Eric Crestan, Arkady Borkovsky, Ana-Maria Popescu, and Vishnu Vyas. 2009. Web-scale distributional similarity and entity set expansion. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, EMNLP '09, pages 938–947, Stroudsburg, PA, USA. Association for Computational Linguistics.

Michael J. Paul, ChengXiang Zhai, and Roxana Girju. 2010. Summarizing contrastive viewpoints in opinionated text. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 66–76, Stroudsburg, PA, USA. Association for Computational Linguistics.

Dafna Shahaf, Carlos Guestrin, and Eric Horvitz. 2012. Trains of thought: Generating information maps. In *International World Wide Web Conference (WWW)*.

Ruben Sipos and Thorsten Joachims. 2013. Generating comparative summaries from reviews. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, pages 1853–1856.

EllenM. Voorhees. 1994. Query expansion using lexical-semantic relations. In BruceW. Croft and C.J. Rijsbergen, editors, *SIGIR ?94*, pages 61–69. Springer London.

Koji Yatani, Michael Novati, Andrew Trusty, and Khai N. Truong. 2011. Review spotlight: A user interface for summarizing user-generated reviews using adjective-noun word pairs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1541–1550, New York, NY, USA. ACM.

Sun Yuepeng, Liu Min, and Wu Cheng. 2011. A modified k-means algorithm for clustering problem with balancing constraints. In *Proceedings of the 2011 Third International Conference on Measuring Technology and Mechatronics Automation - Volume 01*, ICMTMA '11, pages 127–130, Washington, DC, USA. IEEE Computer Society.

ChengXiang Zhai, Atulya Velivelli, and Bei Yu. 2004. A cross-collection mixture model for comparative text mining. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 743–748, New York, NY, USA. ACM.

Shunzhi Zhu, Dingding Wang, and Tao Li. 2010. Data clustering with size constraints. *Knowledge-Based Systems*, 23(8):883 – 889.