Multi-Granular Aspect Aggregation in Aspect-Based Sentiment Analysis

John Pavlopoulos and **Ion Androutsopoulos**
Department of Informatics
Athens University of Economics and Business
Patission 76, GR-104 34 Athens, Greece
http://nlp.cs.aueb.gr/

Abstract

Aspect-based sentiment analysis estimates the sentiment expressed for each particular aspect (e.g., battery, screen) of an entity (e.g., smartphone). Different words or phrases, however, may be used to refer to the same aspect, and similar aspects may need to be aggregated at coarser or finer granularities to fit the available space or satisfy user preferences. We introduce the problem of aspect aggregation at multiple granularities. We decompose it in two processing phases, to allow previous work on term similarity and hierarchical clustering to be reused. We show that the second phase, where aspects are clustered, is almost a solved problem, whereas further research is needed in the first phase, where semantic similarity measures are employed. We also introduce a novel sense pruning mechanism for WordNet-based similarity measures, which improves their performance in the first phase. Finally, we provide publicly available benchmark datasets.

1 Introduction

Given a set of texts discussing a particular entity (e.g., reviews of a laptop), *aspect-based sentiment analysis* (ABSA) attempts to identify the most prominent (e.g., frequently discussed) aspects of the entity (e.g., battery, screen) and the average sentiment (e.g., 1 to 5 stars) for each aspect or group of aspects, as in Fig. 1. Most ABSA systems perform all or some of the following (Liu, 2012): *subjectivity detection* to retain only sentences (or other spans) expressing subjective opinions; *aspect extraction* to extract (and possibly rank) terms corresponding to aspects (e.g., 'battery'); *aspect aggregation* to group aspect terms that are near-synonyms (e.g., 'price', 'cost') or to obtain aspects

Figure 1: Aspect groups and scores of an entity.

at a coarser granularity (e.g., 'chicken','steak', and 'fish' may be replaced by 'food' in restaurant reviews); and *aspect sentiment score estimation* to estimate the average sentiment for each aspect or group of aspects. In this paper, we focus on aspect aggregation, the least studied stage of the four.

Aspect aggregation is needed to avoid reporting separate sentiment scores for aspect terms that are very similar. In Fig. 1, for example, showing separate lines for 'money', 'price', and 'cost' would be confusing. The extent to which aspect terms should be aggregated, however, also depends on the available space and user preferences. On devices with smaller screens, it may be desirable to aggregate aspect terms that are similar, though not necessarily near-synonyms (e.g., 'design', 'color', 'feeling') to show fewer lines (Fig. 1), but finer aspects may be preferable on larger screens. Users may also wish to adjust the granularity of aspects, e.g., by stretching or narrowing the height of Fig. 1 on a smartphone to view more or fewer lines. Hence, aspect aggregation should be able to produce groups of aspect terms for *multiple granularities*. We assume that the aggregated aspects are displayed as lists of terms, as in Fig. 1. We make no effort to order (e.g., by frequency) the terms in each list, nor do we attempt to produce a single (more general) term to describe each aggregated aspect, leaving such tasks for future work.

ABSA systems usually group synonymous (or near-synonymous) aspect terms (Liu, 2012). Ag-

Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pages 78–87,
Gothenburg, Sweden, April 26-30 2014.

gregating only synonyms (or near-synonyms), however, does not allow users to select the desirable aspect granularity, and ignores the hierarchical relations between aspect terms. For example, 'pizza' and 'steak' are kinds of 'food' and, hence, the three terms can be aggregated to show fewer, coarser aspects, even though they are not synonyms. Carenini et al. (2005) used a predefined domain-specific taxonomy to hierarchically aggregate aspect terms, but taxonomies of this kind are often not available. By contrast, we use only general-purpose taxonomies (e.g., WordNet), term similarity measures based on general-purpose taxonomies or corpora, and hierarchical clustering.

We define *multi-granular aspect aggregation* to be the task of partitioning a given set of aspect terms (generated by a previous aspect extraction stage) into k non-overlapping clusters, for multiple values of k. A further constraint is that the clusters have to be *consistent* for different k values, meaning that if two aspect terms t_1, t_2 are placed in the same cluster for $k = k_1$, then t_1 and t_2 must also be grouped together (in the same cluster) for every $k = k_2$ with $k_2 < k_1$, i.e., for every coarser grouping. For example, if 'waiter' and 'service' are grouped together for $k = 5$, they must also be grouped together for $k = 4, 3, 2$ and (trivially) $k = 1$, to allow the user to feel that selecting a smaller number of aspect groups (narrowing the height of Fig. 1) has the effect of zooming out (without aspect terms jumping unexpectedly to other aspect groups), and similarly for zooming in.[1] This requirement is satisfied by using agglomerative hierarchical clustering algorithms (Manning and Schütze, 1999; Hastie et al., 2001), which in our case produce term hierarchies like the ones of Fig. 2. By using slices (nodes at a particular depth) of the hierarchies that are closer to the root or the leaves, we obtain fewer or more clusters. The vertical dotted lines of Fig. 2 illustrate two slices for $k = 4$. By contrast, flat clustering algorithms (e.g., k-means) do not satisfy the consistency constraint for different k values.

Agglomerative clustering algorithms require a measure of the distance between individuals, in our case a measure of how similar two aspect terms are, and a linkage criterion to specify which clusters should be merged to form larger (coarser) clusters. To experiment with different term similarity measures and linkage criteria, we decompose multi-granular aspect aggregation in two processing phases. Phase A fills in a symmetric matrix, like the one of Table 1, with scores showing the similarity of each pair of input aspect terms; the matrix in effect defines the distance measure to be used by agglomerative clustering. In Phase B, the aspect terms are grouped into k non-overlapping clusters, for varying values of k, given the matrix of Phase A and a linkage criterion; a hierarchy like the ones of Fig. 2 is first formed via agglomerative clustering, and fewer or more clusters (for different values of k) are then obtained by using different slices of the hierarchy, as already discussed. Our two-phase decomposition can also accommodate non-hierarchical clustering algorithms, provided that the consistency constraint is satisfied, but we consider only agglomerative hierarchical clustering in this paper.

The decomposition in two phases has three main advantages. Firstly, it allows reusing previous work on term similarity measures (Zhang et al., 2013), which can be used to fill in the matrix of Phase A. Secondly, the decomposition allows different linkage criteria to be experimentally compared (in Phase B) using the same similarity matrix (of Phase A), i.e., the same distance

Figure 2: Example aspect hierarchies produced by agglomerative hierarchical clustering.

	food	*fish*	*sushi*	*dishes*	*wine*
food	5	4	4	4	2
fish	4	5	4	2	1
sushi	4	4	5	3	1
dishes	4	2	3	5	2
wine	2	1	1	2	5

Table 1: An aspect term similarity matrix.

[1] We also require the clusters to be non-overlapping to make this zooming in and out metaphor clearer to the user.

measure. Thirdly, the decomposition leads to high inter-annotator agreement, as we show experimentally. By contrast, in preliminary experiments we found that asking humans to directly evaluate aspect hierarchies produced by hierarchical clustering, or to manually create gold aspect hierarchies led to poor inter-annotator agreement.

We show that existing term similarity measures perform reasonably well in Phase A, especially when combined, but there is a large scope for improvement. We also propose a novel *sense pruning* method for WordNet-based similarity measures, which leads to significant improvements in Phase A. In Phase B, we experiment with agglomerative clustering using four different linkage criteria, concluding that they all perform equally well and that Phase B is almost a solved problem when the gold similarity matrix of Phase A is used; however, further improvements are needed in the similarity measures of Phase A to produce a sufficiently good similarity matrix. We also make publicly available the datasets of our experiments.

Our main contributions are: (i) to the best of our knowledge, we are the first to consider multi-granular aspect aggregation (not just merging near-synonyms) in ABSA *without* manually crafted domain-specific ontologies; (ii) we propose a two-phase decomposition that allows previous work on term similarity and hierarchical clustering to be reused and evaluated with high inter-annotator agreement; (iii) we introduce a novel sense pruning mechanism that improves WordNet-based similarity measures; (iv) we provide the first public datasets for multi-granular aspect aggregation; (v) we show that the second phase of our decomposition is almost a solved problem, and that research should focus on the first phase. Although we experiment with customer reviews of products and services, ABSA and the work of this paper in particular are, at least in principle, also applicable to texts expressing opinions about other kinds of entities (e.g., politicians, organizations).

Section 2 below discusses related work. Sections 3 and 4 present our work for Phase A and B, respectively. Section 5 concludes.

2 Related work

Most existing approaches to aspect aggregation aim to produce a single, *flat* partitioning of aspect terms into aspect groups, rather than aspect groups at multiple granularities. The most common approaches (Liu, 2012) are to aggregate only synonyms or near-synonyms, using WordNet (Liu et al., 2005), statistics from corpora (Chen et al., 2006; Bollegala et al., 2007a; Lin and Wu, 2009), or semi-supervised learning (Zhai et al., 2010; Zhai et al., 2011), or to cluster the aspect terms using (latent) topic models (Titov and McDonald, 2008a; Guo et al., 2009; Brody and Elhadad, 2010; Jo and Oh, 2011). Topic models do not perform better than other methods (Zhai et al., 2010), and their clusters may overlap.[2] The topic model of Titov et al. (2008b) uses two granularity levels; we consider many more (3–10 levels).

Carenini et al. (2005) used a *predefined domain-specific* taxonomy and similarity measures to aggregate related terms. Yu et al. (2011) used a tailored version of an existing taxonomy. By contrast, we assume no domain-specific taxonomy. Kobayashi et al. (2007) proposed methods to extract aspect terms and relations between them, including hierarchical relations. They extract, however, relations by looking for clues in texts (e.g., particular phrases). By contrast, we employ similarity measures and hierarchical clustering, which allows us to group similar aspect terms even when they do not cooccur in texts. Also, in contrast to Kobayashi et al. (2007), we respect the consistency constraint discussed in Section 1.

A similar task is taxonomy induction. Cimiano and Staab (2005) automatically construct taxonomies from texts via agglomerative clustering, much as in our Phase B, but not in the context of ABSA, and without trying to learn a similarity matrix first. They also label the hierarchy's concepts, a task we do not consider. Klapaftis and Manandhar (2010) show how word sense induction can be combined with agglomerative clustering to obtain more accurate taxonomies, again not in the context of ABSA. Our sense pruning method was influenced by their work, but is much simpler than their word sense induction. Fountain and Lapata (2012) study unsupervised methods to induce concept taxonomies, without considering ABSA.

3 Phase A

We now discuss our work for Phase A. Recall that in this phase the input is a set of aspect terms and

[2] Topic models are typically also used to perform aspect extraction, apart from aspect aggregation, but simple heuristics (e.g., most frequent nouns) often outperform them in aspect extraction (Liu, 2012; Moghaddam and Ester, 2012).

the goal is to fill in a matrix (Table 1) with scores showing the similarity of each pair of aspect terms.

3.1 Datasets used in Phase A

We used two benchmark datasets that we had previously constructed to evaluate ABSA methods for subjectivity detection, aspect extraction, and aspect score estimation, but not aspect aggregation. We extended them to support aspect aggregation, and we make them publicly available.[3]

The two original datasets contain sentences from customer reviews of restaurants and laptops, respectively. The reviews are manually split into sentences, and each sentence is manually annotated as 'subjective' (expressing opinion) or 'objective' (not expressing opinion). The restaurants dataset contains 3,710 English sentences from the restaurant reviews of Ganu et al. (2009). The laptops dataset contains 3,085 English sentences from 394 customer reviews, collected from sites that host customer reviews. In the experiments of this paper, we use only the 3,057 (out of 3,710) subjective restaurant sentences and the 2,631 (out of 3,085) subjective laptop sentences.

For each subjective sentence, our datasets show the words that human annotators marked as aspect terms. For example, in "The *dessert* was divine!" the aspect term is 'dessert', and in "Really bad *waiter*." it is 'waiter'. Among the 3,057 subjective restaurant sentences, 1,129 contain exactly one aspect term, 829 more than one, and 1,099 no aspect term; a subjective sentence may express an opinion about the restaurant (or laptop) being reviewed without mentioning a specific aspect (e.g., "Really nice restaurant!"), which is why no aspect terms are present in some subjective sentences. There are 558 distinct multi-word aspect terms and 431 distinct single-word aspect terms in the subjective restaurant sentences. Among the 2,631 subjective sentences of the laptop reviews, 823 contain exactly one aspect term, 389 more than one, and 1,419 no aspect term. There are 273 distinct multi-word aspect terms and 330 distinct single-word aspect terms in the subjective laptop sentences.

From each dataset, we selected the 20 (distinct) aspect terms that the human annotators had annotated most frequently, taking annotation frequency to be an indicator of importance; there are only two multi-word aspect terms ('hard drive', 'battery life') among the 20 most frequent ones in the laptops dataset, and none among the 20 most frequent aspect terms of the restaurants dataset. We then formed all the 190 possible pairs of the 20 terms and constructed an empty similarity matrix (Fig. 1), one for each dataset, which was given to three human judges to fill in (1: strong dissimilarity, 5: strong similarity).[4] For each aspect term, all the subjective sentences mentioning the term were also provided, to help the judges understand how the terms are used in the particular domains (e.g., 'window' and 'Windows' have domain-specific meanings in laptop reviews).

The Pearson correlation coefficient indicated high inter-annotator agreement (0.81 for restaurants, 0.74 for laptops). We also measured the absolute inter-annotator agreement $a(l_1, l_2)$, defined below, where l_1, l_2 are lists containing the scores (similarity matrix values) of two judges, N is the length of each list, and v_{max}, v_{min} are the largest and smallest possible scores (5 and 1).

$$a(l_1, l_2) = \frac{1}{N} \sum_{i=1}^{N} \left[1 - \frac{|l_1(i) - l_2(i)|}{v_{max} - v_{min}} \right]$$

The absolute interannotator agreement was also high (0.90 for restaurants, 0.91 for laptops).[5] With both measures, we compute the agreement of each judge with the averaged (for each matrix cell) scores of the other two judges, and we report the mean of the three agreement estimates. Finally, we created the *gold* similarity matrix of each dataset by placing in each cell the average scores that the three judges had provided for that cell.

In preliminary experiments, we gave aspect terms to human judges, asking them to group any terms they considered near-synonyms. We then asked the judges to group the aspect terms into fewer, coarser groups by grouping terms that could be viewed as direct hyponyms of the same broader term (e.g., 'pizza' and 'steak' are both kinds of 'food'), or that stood in a hyponym-hypernym relation (e.g., 'pizza' and 'food'). We used the Dice coefficient to measure inter-annotator agreement, and we obtained reasonably good agreement for near-synonyms (0.77 for restaurants, 0.81 for laptops), but poor agreement for the coarser as-

[3] The datasets are available at `http://nlp.cs.aueb.gr/software.html`.

[4] The matrix is symmetric; hence, the judges had to fill in only half of it. The guidelines and an annotation tool that were given to the judges are available upon request.

[5] The Pearson correlation ranges from -1 to 1, whereas the absolute inter-annotator agreement ranges from 0 to 1.

pects (0.25 and 0.11).[6] In other preliminary experiments, we asked human judges to rank alternative aspect hierarchies that had been produced by applying agglomerative clustering with different linkage criteria to 20 aspect terms, but we obtained very poor inter-annotator agreement (Pearson score -0.83 for restaurants and 0 for laptops).

3.2 Phase A methods

We employed five term similarity measures. The first two are WordNet-based (Budanitsky and Hirst, 2006). The next two combine WordNet with statistics from corpora. The fifth one is a corpus-based distributional similarity measure.

The first measure is *Wu and Palmer's* (1994). It is actually a sense similarity measure (a term may have multiple senses). Given two senses $s_{ij}, s_{i'j'}$ of terms $t_i, t_{i'}$, the measure is defined as follows:

$$WP(s_{ij}, s_{i'j'}) = 2 \cdot \frac{depth(lcs(s_{ij}, s_{i'j'}))}{depth(s_{ij}) + depth(s_{ij})},$$

where $lcs(s_{ij}, s_{i'j'})$ is the *least common subsumer*, i.e., the most specific common ancestor of the two senses in WordNet, and $depth(s)$ is the depth of sense s in WordNet's hierarchy.

Most terms have multiple senses, however, and word sense disambiguation methods (Navigli, 2009) are not yet robust enough. Hence, when given two aspect terms $t_i, t_{i'}$, rather than particular senses of the terms, a simplistic *greedy* approach is to compute the similarities of all the possible pairs of senses $s_{ij}, s_{i'j'}$ of $t_i, t_{i'}$, and take the similarity of $t_i, t_{i'}$ to be the maximum similarity of the sense pairs (Bollegala et al., 2007b; Zesch and Gurevych, 2010). We use this greedy approach with all the WordNet-based measures, but we also propose a sense pruning mechanism below, which improves their performance. In all the WordNet-based measures, if a term is not in WordNet, we take its similarity to any other term to be zero.[7]

The second measure, $PATH(s_{ij}, s_{i'j'})$, is simply the inverse of the length (plus one) of the shortest path connecting the senses $s_{ij}, s_{i'j'}$ in WordNet (Zhang et al., 2013). Again, the greedy approach can be used with terms having multiple senses.

The third measure is *Lin's* (1998), defined as:

$$LIN(s_{ij}, s_{i'j'}) = \frac{2 \cdot ic(lcs(s_{ij}, s_{i'j'}))}{ic(s_{ij}) + ic(s_{i'j'})},$$

where $s_{ij}, s_{i'j'}$ are senses of terms $t_i, t_{i'}$, $lcs(s_{ij}, s_{i'j'})$ is the least common subsumer of $s_{ij}, s_{i'j'}$ in WordNet, and $ic(s) = -\log P(s)$ is the *information content* of sense s (Pedersen et al., 2004), estimated from a corpus. When the corpus is not sense-tagged, we follow the common approach of treating each occurrence of a word as an occurrence of all of its senses, when estimating $ic(s)$.[8] We experimented with two variants of Lin's measure, one where the $ic(s)$ scores were estimated from the Brown corpus (Marcus et al., 1993), and one where they were estimated from the (restaurant or laptop) reviews of our datasets.

The fourth measure is *Jiang and Conrath's* (1997), defined below. Again, we experimented with two variants of $ic(s)$, as above.

$$JCN(s_{ij}, s_{i'j'}) = \frac{1}{ic(s_{ij}) + ic(s_{i'j'}) - 2 \cdot lcs(s_{ij}, s_{i'j'})}$$

For all the above WordNet-based measures, we experimented with a *sense pruning* mechanism, which discards some of the senses of the aspect terms, before applying the greedy approach. For each aspect term t_i, we consider all of its WordNet senses s_{ij}. For each s_{ij} and each other aspect term $t_{i'}$, we compute (using $PATH$) the similarity between s_{ij} and each sense $s_{i'j'}$ of $t_{i'}$, and we consider the *relevance* of s_{ij} to $t_{i'}$ to be:[9]

$$rel(s_{ij}, t_{i'}) = \max_{s_{i'j'} \in senses(t_{i'})} PATH(s_{ij}, s_{i'j'})$$

The relevance of s_{ij} to *all* of the N other aspect terms $t_{i'}$ is taken to be:

$$rel(s_{ij}) = \frac{1}{N} \cdot \sum_{i' \neq i} rel(s_{ij}, t_{i'})$$

For each aspect term t_i, we retain only its senses s_{ij} with the top $rel(s_{ij})$ scores, which tends to

[6] The Dice coefficient ranges from 0 to 1. There was a very large number of possible responses the judges could provide and, hence, it would be inappropriate to use Cohen's K.

[7] This never happened in the restaurants dataset. In the laptops dataset, it only happened for 'hard drive' and 'battery life'. We use the NLTK implementation of the first four measures (see `http://nltk.org/`) and our own implementation of the distributional similarity measure.

[8] `http://www.d.umn.edu/~tpederse/Data/README-WN-IC-30.txt`. We use the default counting.

[9] We also experimented with other similarity measures when computing $rel(s_{ij}, t_{i'})$, instead of $PATH$, but there was no significant difference. We use NLTK to tokenize, remove punctuation, and stop-words.

	without SP		with SP	
Method	*Rest.*	*Lapt.*	*Rest.*	*Lapt.*
WP	0.475	0.216	0.502	0.265
$PATH$	**0.524**	0.301	0.529	0.332
$LIN@domain$	0.390	0.256	0.456	0.343
$LIN@Brown$	0.434	0.329	0.471	0.391
$JCN@domain$	0.467	0.348	0.509	0.448
$JCN@Brown$	0.403	0.469	0.419	0.539
DS	0.283	**0.517**	(0.283)	(0.517)
AVG	0.499	0.352	0.537	0.426
WN	0.490	0.328	0.530	0.395
$WNDS$	0.523	0.453	**0.545**	**0.546**

Table 2: Phase A results (Pearson correlation to gold similarities) *with* and *without* sense pruning.

prune senses that are very irrelevant to the particular domain (e.g., laptops). This sense pruning mechanism is novel, and we show experimentally that it improves the performance of all the WordNet-based similarity measures we examined.

We also implemented a *distributional similarity* measure (Harris, 1968; Padó and Lapata, 2007; Cimiano et al., 2009; Zhang et al., 2013). Following Lin and Wu (2009), for each aspect term t, we create a vector $\vec{v}(t) = \langle PMI(t, w_1), \ldots, PMI(t, w_n)\rangle$. The vector components are the Pointwise Mutual Information scores of t and each word w_i of a corpus:

$$PMI(t, w_i) = -\log \frac{P(t, w_i)}{P(t) \cdot P(w_i)}$$

We treat $P(t, w_i)$ as the probability of t, w_i cooccurring in the same sentence, and we use the (laptop or restaurant) reviews of our datasets as the corpus to estimate the probabilities. The distributional similarity $DS(t, t')$ of two aspect terms t, t' is the cosine similarity of $\vec{v}(t), \vec{v}(t')$.[10]

Finally, we tried combinations of the similarity measures: AVG is the average of all five; WN is the average of the first four, which employ WordNet; and $WNDS$ is the average of WN and DS; all the scores range in $[0, 1]$. We also tried regression (e.g., SVR), but there was no improvement.

3.3 Phase A experimental results

Each similarity measure was evaluated by computing its Pearson correlation with the scores of the gold similarity matrix. Table 2 shows the results.

Our sense pruning consistently improves all four WordNet-based measures. It does not apply to DS, which is why the DS results are identical with and without pruning. A paired t test indicates that the other differences (with and without pruning) of Table 2 are statistically significant ($p < 0.05$). We used the senses with the top five $rel(s_{ij})$ scores for each aspect term t_i during sense pruning. We also experimented with keeping fewer senses, but the results were inferior or there was no improvement.

Lin's measure performed better when information content was estimated on the (much larger, but domain-independent) Brown corpus ($LIN@Brown$), as opposed to using the (domain-specific) reviews of our datasets ($LIN@domain$), but we observed no similar consistent pattern for JCN. Given its simplicity, $PATH$ performed remarkably well in the restaurants dataset; it was the best measure (including combinations) without sense pruning, and the best uncombined measure with sense pruning. It performed worse, however, compared to several other measures in the laptops dataset. Similar comments apply to WP, which is among the top-performing uncombined measures in restaurants, both with and without sense pruning, but the worst overall measure in laptops. DS is the best overall measure in laptops when compared to measures without sense pruning, and the third best overall when compared to measures that use sense pruning, but the worst overall in restaurants both with and without pruning. LIN and JCN, which use both WordNet and corpus statistics, have a more balanced performance across the two datasets, but they are not top-performers in any of the two. *Combinations of similarity measures seem more stable across domains*, as the results of AVG, WN, and $WNDS$ indicate, though experiments with more domains are needed to investigate this issue. *$WNDS$ is the best overall method with sense pruning*, and among the best three methods without pruning in both datasets.

To get a better view of the performance of $WNDS$ with sense pruning, i.e., the best overall measure of Table 2, we compared it to two state of the art semantic similarity systems. First, we applied the system of Han et al. (2013), one of the best systems of the recent *Sem 2013 semantic text similarity competition, to our Phase A data. The performance (Pearson correlation with gold similarities) of the same system on the widely used *WordSim353* word similarity dataset (Agirre et al., 2009) is 0.73, much higher than the same system's performance on our Phase A data (see Table 3),

[10] We also experimented with Euclidean distance, a normalized PMI (Bouma, 2009), and the Brown corpus, but there was no improvement.

Method	Restaurants	Laptops
Han et al. (2013)	0.450	0.471
Word2Vec	0.434	0.485
WNDS with SP	**0.545**	**0.546**
Judge 1	0.913	0.875
Judge 2	0.914	0.894
Judge 3	0.888	0.924

Table 3: Phase A results (Pearson correlation to gold similarities) of *WNDS* with SP against semantic similarity systems and human judges.

which suggests that our data are more difficult.[11]

We also employed the recent *Word2Vec* system, which computes continuous vector space representations of words from large corpora and has been reported to improve results in word similarity tasks (Mikolov et al., 2013). We used the English Wikipedia to compute word vectors with 200 features.[12] The similarity between two aspect terms was taken to be the cosine similarity of their vectors. This system performed better than Han et al.'s with laptops, but not with restaurants.

Table 3 shows that *WNDS (with sense pruning) performed clearly better than the system of Han et al. and Word2Vec*. Table 3 also shows the Pearson correlation of each judge's scores to the gold similarity scores, as an indication of the best achievable results. Although *WNDS* (with sense pruning) performs reasonably well in both domains,[13] *there is large scope for improvement.*

4 Phase B

In Phase B, the aspect terms are to be grouped into k non-overlapping clusters, for varying values of k, given a Phase A similarity matrix. We experimented with both the gold similarity matrix of Phase A and similarity matrices produced by *WNDS* (with SP), the best Phase A method.

4.1 Phase B methods

We experimented with agglomerative clustering and four linkage criteria: *single*, *complete*, *average*, and *Ward* (Manning and Schütze, 1999; Hastie et al., 2001). Let $d(t_1, t_2)$ be the distance of two individual instances t_1, t_2; in our case, the instances are aspect terms and $d(t_1, t_2)$ is the inverse of the similarity of t_1, t_2, defined by the Phase A similarity matrix (gold or produced by *WNDS*). Different linkage criteria define differently the distance of two clusters $D(C_1, C_2)$, which affects the choice of clusters that are merged to produce coarser (higher-level) clusters:

$$D_{single}(C_1, C_2) = \min_{t_1 \in C_1, t_2 \in C_2} d(t_1, t_2)$$

$$D_{compl}(C_1, C_2) = \max_{t_1 \in C_1, t_2 \in C_2} d(t_1, t_2)$$

$$D_{avg}(C_1, C_2) = \frac{1}{|C_1||C_2|} \sum_{t_1 \in C_1} \sum_{t_2 \in C_2} d(t_1, t_2)$$

Complete linkage tends to produce more compact clusters, compared to single linkage, with average linkage being in between. Ward minimizes the total in-cluster variance; consult Milligan (1980) for further details.[14]

4.2 Phase B experimental results

To evaluate the k clusters produced at each aspect granularity by the different linkage criteria, we used the *Silhouette Index* (*SI*) (Rousseeuw, 1987), a cluster evaluation measure that considers both inter- and intra-cluster coherence.[15] Given a set of clusters $\{C_1, \ldots, C_k\}$, each $SI(C_i)$ is defined as:

$$SI(C_i) = \frac{1}{|C_i|} \cdot \sum_{j=1}^{|C_i|} \frac{b_j - a_j}{\max(b_j, a_j)},$$

where a_j is the mean distance from the j-th instance of C_i to the other instances in C_i, and b_j is the mean distance from the j-th instance of C_i to the instances in the cluster nearest to C_i. Then:

$$SI(\{C_1, \ldots, C_k\}) = \frac{1}{k} \cdot \sum_{i=1}^{k} SI(C_i)$$

We always use the correct (gold) distances of the instances (terms) when computing the *SI* scores.

As shown in Fig. 3, *no linkage criterion clearly outperforms the others, when the gold matrix of Phase A is used*; all four criteria perform reasonably well. Note that the *SI* ranges from -1 to

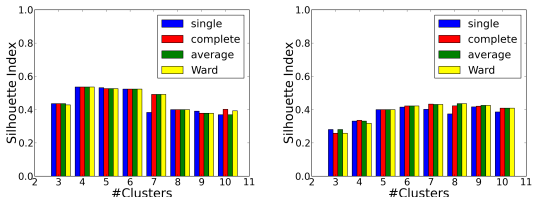
[11] The system of Han et al. (2013) is available from `http://semanticwebarchive.cs.umbc.edu/SimService/`; we use the STS similarity.

[12] *Word2Vec* is available from `https://code.google.com/p/word2vec/`. We used the continuous bag of words model with default parameters, the first billion characters of the English Wikipedia, and the preprocessing of `http://mattmahoney.net/dc/textdata.html`.

[13] Recall that the Pearson correlation ranges from -1 to 1.

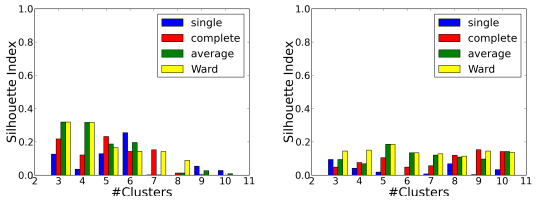
[14] We used the SciPy implementations of agglomerative clustering with the four criteria (see `http://www.scipy.org`), relying on *maxclust* to obtain the slice of the resulting hierarchy that leads to k (or approx. k) clusters.

[15] We used the *SI* implementation of Pedregosa et al. (2011); see `http://scikit-learn.org/`. We also experimented with the Dunn Index (Dunn, 1974) and the Davies-Bouldin Index (1979), but we obtained similar results.



(a) restaurants (b) laptops

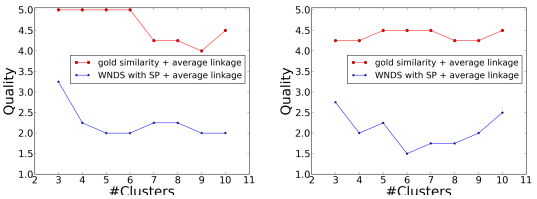
Figure 3: Silhouette Index (SI) results for Phase B, using the **gold** similarity matrix of Phase A.



(a) restaurants (b) laptops

Figure 4: SI results for Phase B, using the **WNDS (with SP)** similarity matrix of Phase A.

1, with higher values indicating better clustering. Figure 4 shows that *when the similarity matrix of WNDS (with SP) is used, the SI scores deteriorate significantly*; again, there is no clear winner among the linkage criteria, but average and Ward seem to be overall better than the others.



(a) Restaurants (b) Laptops

Figure 5: Human evaluation of aspect groups.

In a final experiment, we showed clusterings of varying granularities (k values) to four human judges (graduate CS students). The clusterings were produced by two systems: one that used the *gold similarity matrix* of Phase A and agglomerative clustering with average linkage in Phase B, and one that used the *similarity matrix of WNDS* (with SP) and again agglomerative clustering with average linkage. We showed all the clusterings to all the judges. Each judge was asked to evaluate each clustering on a 1–5 scale. We measured the absolute inter-annotator agreement, as in Section 3.1, and found high agreement in all cases (0.93 and 0.83 for the two systems, respectively, in restaurants; 0.85 for both in laptops).[16]

Figure 5 shows the average human scores of the two systems for different granularities. The judges considered the aspect groups always perfect or near-perfect when the gold similarity matrix of Phase A was used, but they found the aspect groups to be of rather poor quality when the similarity matrix of the best Phase A measure was used. These results, along with those of Fig. 3–4, show that *more effort needs to be devoted to improving the similarity measures of Phase A, whereas Phase B is in effect an almost solved problem*, if a good similarity matrix is available.

5 Conclusions

We considered a new, more demanding form of aspect aggregation in ABSA, which aims to aggregate aspects at multiple granularities, as opposed to simply merging near-synonyms, and without assuming that manually crafted domain-specific ontologies are available. We decomposed the problem in two processing phases, which allow previous work on term similarity and hierarchical clustering to be reused and evaluated appropriately with high inter-annotator agreement. We showed that the second phase, where we used agglomerative clustering, is an almost solved problem, whereas further research is needed in the first phrase, where term similarity measures are employed. We also introduced a sense pruning mechanism that significantly improves WordNet-based similarity measures, leading to a measure that outperforms state of the art similarity methods in the first phase of our decomposition. We also made publicly available the datasets of our experiments.

Acknowledgments

We thank G. Batistatos, A. Zosakis, and G. Lampouras for their annotations in Phase A. We thank A. Kosmopoulos, G. Lampouras, P. Malakasiotis, and I. Lourentzou for their annotations in Phase B.

[16] The Pearson correlation cannot be computed, as several judges gave the same rating to the first system, for all k.

References

E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Paşca, and A. Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of the Annual Conference of NAACL*, pages 19–27, Boulder, CO, USA.

D. Bollegala, Y. Matsuo, and M. Ishizuka. 2007a. An integrated approach to measuring semantic similarity between words using information available on the web. In *Proceedings of HLT-NAACL*, pages 340–347, Rochester, NY, USA.

D. Bollegala, Y. Matsuo, and M. Ishizuka. 2007b. Measuring semantic similarity between words using web search engines. In *Proceedings of the 16th International Conference of WWW*, volume 766, pages 757–766, Banff, Alberta, Canada.

G. Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of the Biennial Conference of GSCL*, pages 31–40.

S. Brody and N. Elhadad. 2010. An unsupervised aspect-sentiment model for online reviews. In *Proceedings of the Annual Conference of NAACL*, pages 804–812, Los Angeles, CA, USA.

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

G. Carenini, R. T. Ng, and E. Zwart. 2005. Extracting knowledge from evaluative text. In *Proceedings of the 3rd International Conference on Knowledge Capture*, pages 11–18, Banff, Alberta, Canada.

H. Chen, M. Lin, and Y. Wei. 2006. Novel association measures using web search with double checking. In *Proceedings of the 21st International Conference of COLING and the 44th Annual Meeting of ACL*, pages 1009–1016, Sydney, Australia.

P. Cimiano and S. Staab. 2005. Learning concept hierarchies from text with a guided hierarchical clustering algorithm. In *Proceedings of ICML – Workshop on Learning and Extending Lexical Ontologies with Machine Learning Methods*, Bonn, Germany.

P. Cimiano, A. Mädche, S. Staab, and J. Völker. 2009. Ontology learning. In *Handbook on Ontologies*, pages 245–267. Springer.

D. L. Davies and D. W. Bouldin. 1979. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(2):224–227.

J. C. Dunn. 1974. Well-separated clusters and optimal fuzzy partitions. *Journal of Cybernetics*, 4(1):95–104.

T. Fountain and M. Lapata. 2012. Taxonomy induction using hierarchical random graphs. In *Proceedings of NAACL:HLT*, pages 466–476, Montreal, Canada.

G. Ganu, N. Elhadad, and A. Marian. 2009. Beyond the stars: Improving rating predictions using review text content. In *Proceedings of the 12th International Workshop on the Web and Databases*, Providence, RI, USA.

H. Guo, H. Zhu, Z. Guo, X. Zhang, and Z. Su. 2009. Product feature categorization with multilevel latent semantic association. In *Proceedings of the 18th CIKM*, pages 1087–1096.

L. Han, A. Kashyap, T. Finin, J. Mayfield, and J. Weese. 2013. Umbc_ebiquity-core: Semantic textual similarity systems. In *Proceedings of the 2nd Joint Conference on Lexical and Computational Semantics*, pages 44–52, Atlanta, GA, USA.

Z. Harris. 1968. *Mathematical Structures of Language*. Wiley.

T. Hastie, R. Tibshirani, and J. Friedman. 2001. *The Elements of Statistical Learning*. Springer.

J. J. Jiang and D. W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of ROCLING*, pages 19–33, Taiwan, China.

Y. Jo and A. H. Oh. 2011. Aspect and sentiment unification model for online review analysis. In *Proceedings of the 4th International Conference of WSDM*, pages 815–824, Hong Kong, China.

I. P. Klapaftis and S. Manandhar. 2010. Taxonomy learning using word sense induction. In *Proceedings of NAACL*, pages 82–90, Los Angeles, CA, USA.

N. Kobayashi, K. Inui, and Y. Matsumoto. 2007. Extracting aspect-evaluation and aspect-of relations in opinion mining. In *Proceedings of the Joint Conference on EMNLP-CoNLL*, pages 1065–1074, Prague, Czech Republic.

D. Lin and X. Wu. 2009. Phrase clustering for discriminative learning. In *Proceedings of ACL*, pages 1030–1038, Suntec, Singapore. ACL.

D. Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the 15th ICML*, pages 296–304, Madison, WI, USA.

B. Liu, M. Hu, and J. Cheng. 2005. Opinion observer: analyzing and comparing opinions on the web. In *Proceedings of the 14th International Conference of WWW*, pages 342–351, Chiba, Japan.

B. Liu. 2012. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool.

C. D. Manning and H. Schütze. 1999. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, USA.

M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini. 1993. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2):313–330.

T. Mikolov, C. Kai, G. Corrado, and J. Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

G.W. Milligan. 1980. An examination of the effect of six types of error perturbation on fifteen clustering algorithms. *Psychometrika*, 45(3):325–342.

S. Moghaddam and M. Ester. 2012. On the design of lda models for aspect-based opinion mining. In *Proceedings of the 21st CIKM*, pages 803–812, Maui, HI, USA.

R. Navigli. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2):10:1–10:69.

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

T. Pedersen, S. Patwardhan, and J. Michelizzi. 2004. Wordnet::similarity: measuring the relatedness of concepts. In *Proceedings of NAACL:HTL – Demonstrations*, pages 38–41, Boston, MA, USA.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.

P. Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20(1):53–65.

I. Titov and R. T. McDonald. 2008a. A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of the 46th Annual Meeting of ACL-HLT*, pages 308–316, Columbus, OH, USA.

I. Titov and R. T. McDonald. 2008b. Modeling online reviews with multi-grain topic models. In *Proceedings of the 17th International Conference of WWW*, pages 111–120, Beijing, China.

Z. Wu and M. Palmer. 1994. Verbs semantics and lexical selection. In *Proceedings of the 32nd ACL*, pages 133–138, Las Cruces, NM, USA.

J. Yu, Z. Zha, M. Wang, K. Wang, and T. Chua. 2011. Domain-assisted product aspect hierarchy generation: towards hierarchical organization of unstructured consumer reviews. In *Proceedings of EMNLP*, pages 140–150, Edinburgh, UK.

T. Zesch and I. Gurevych. 2010. Wisdom of crowds versus wisdom of linguists - measuring the semantic relatedness of words. *Natural Language Engineering*, 16(1):25–59.

Z. Zhai, B. Liu, H. Xu, and P. Jia. 2010. Grouping product features using semi-supervised learning with soft-constraints. In *Proceedings of the 23rd International Conference of COLING*, pages 1272–1280, Beijing, China.

Z. Zhai, B. Liu, H. Xu, and P. Jia. 2011. Clustering product features for opinion mining. In *Proceedings of the 4th International Conference of WSDM*, pages 347–354, Hong Kong, China.

Z. Zhang, A. Gentile, and F. Ciravegna. 2013. Recent advances in methods of lexical semantic relatedness - a survey. *Natural Language Engineering*, FirstView(1):1–69.