Sense-aware Semantic Analysis: A Multi-prototype Word Representation Model using Wikipedia

Zhaohui Wu[†], **C. Lee Giles**[‡†]
[†]Computer Science and Engineering, [‡]Information Sciences and Technology
Pennsylvania State University, University Park, PA 16802, USA
zzw109@psu.edu, giles@ist.psu.edu

Abstract

Human languages are naturally ambiguous, which makes it difficult to automatically understand the semantics of text. Most vector space models (VSM) treat all occurrences of a word as the same and build a single vector to represent the meaning of a word, which fails to capture any ambiguity. We present sense-aware semantic analysis (*SaSA*), a multi-prototype VSM for word representation based on Wikipedia, which could account for homonymy and polysemy. The "sense-specific" prototypes of a word are produced by clustering Wikipedia pages based on both local and global contexts of the word in Wikipedia. Experimental evaluation on semantic relatedness for both isolated words and words in sentential contexts and word sense induction demonstrate its effectiveness.

Introduction

Computationally modeling semantics of text has long been a fundamental task for natural language understanding. Among many approaches for semantic modeling, distributional semantic models using large scale corpora or web knowledge bases have proven to be effective (Deerwester et al. 1990; Gabrilovich and Markovitch 2007; Mikolov et al. 2013). Specifically, they provide vector embeddings for a single text unit based on the distributional context where it occurs, from which semantic relatedness or similarity measures can be derived by computing distances between vectors. However, a common limitation of most vector space models is that each word is only represented by a single vector, which cannot capture homonymy and polysemy (Reisinger and Mooney 2010). A natural way to address this limitation could be building multi-prototype models that provide different embeddings for different senses of a word. However, this task is under studied with only a few exceptions (Reisinger and Mooney 2010; Huang et al. 2012), which cluster the contexts of a word into K clusters to represent multiple senses.

While these multi-prototype models showed significant improvement over single prototype models, there are two fundamental problems yet to be addressed. First, they simply predefine a fixed number of prototypes, K, for every word in the vocabulary, which should not be the case since different words could have a different number of senses. Second, the sense-specific context clusters are generated from free text corpus, whose quality cannot be guaranteed nor evaluated (Purandare and Pedersen 2004; Schütze 1998). It is possible that contexts of different word senses could be clustered together because they might share some common words, while contexts of the same word sense could be clustered into different groups since they have no common words. For example, apple "Apple Inc." and apple "Apple Corps" share many contextual words in Wikipedia such as "computer", "retail", "shares", and "logs" even if we consider a context window size of only 3.

Thus, the question posed would be how can we build a sense-aware semantic profile for a word that can give accurate sense-specific prototypes in terms of both number and quality? And for a given context of the word, can the model assign the semantic representation of a word that corresponds to the specific sense?

By comparing existing methods that adopted automatic sense induction from free text based on context clustering, a better way to incorporate sense-awareness into semantic modeling is to do word sense disambiguation for different occurrences of a word using manually complied sense inventories such as WordNet (Miller 1995). However, due to *knowledge acquisition bottleneck* (Gale, Church, and Yarowsky 1992b), this approach may often miss corpus/domain-specific senses and may be out of date due to changes in human languages and web content (Pantel and Lin 2002). As such, we will use Wikipedia, the largest encyclopedia knowledge base online with rich semantic information and wide knowledge coverage, as a semantic corpus on which to test our Sense-aware Semantic Analysis *SaSA*. Each dimension in *SaSA* is a Wikipedia concept/article[1] where a word appears or co-occurs with. By assuming that occurrences of a word in Wikipedia articles of similar subjects should share the sense, the sense-specific clusters are generated by agglomerative hierarchical clustering based on not only the text context, but also Wikipedia links and categories that could ensure more semantics, giving different words their own clusters. The links give unique identification of a word occurrence by

[1]Each concept corresponds to a unique Wikipedia article.

linking it to a Wikipedia article which provides helpful local disambiguated information. The categories give global topical labels of a Wikipedia article that could also be helpful for sense induction. For example, while the pure text context of word apple in “Apple Inc.” and “Apple Corps” could not differentiate the two senses, the categories of the two concepts may easily show the difference since they have no category labels in common.

Our contributions can be summarized as follows:

- We propose a multi-prototype model for word representation, namely *SaSA*, using Wikipedia that could give more accurate sense-specific representation of words with multiple senses.
- We apply *SaSA* to different semantic relatedness tasks, including word-to-word (for both isolated words and words in sentential contexts) and text-to-text, and achieve better performance than the state-of-the-art methods in both single prototype and multi-prototype models.

Sense-aware Semantic Analysis

SaSA follows ESA by representing a word using Wikipedia concepts. Given the whole Wikipedia concept set $\mathcal{W} = \{C_1, ..., C_n\}$, a word w, and the concept set that relates to the word $C(w) = \{C_{w_1}, ..., C_{w_k}\}$, *SaSA* models w as its sense-aware semantic vector $V(w_{s_i}) = [r_{i1}(w), ..., r_{ih}(w)]$, where $r_{ij}(w)$ measures the relevance of w under sense s_i to concept C_{ij}, and $S(w) = \{s_1, ..., s_m\}$ denotes all the senses of w inducted from $C(w)$. Specifically, $s_i = \{C_{i1}, ..., C_{ih}\} \subset C(w)$ is a sense cluster containing a set of Wikipedia concepts where occurrences of w share the sense.

Figure 1 demonstrates the work flow of *SaSA*. Given a word w, it first finds all Wikipedia concepts that relate to w, including those contain w (C1, C5, and C7) and those co-occur with w as Wikipedia links in w’s contexts (C2, C3, C4, and C6). We define a context of w as a sentence containing it. Then it uses agglomerative hierarchical clustering to group the concepts sharing the sense of w into a cluster. All the sense clusters represent the sense space $S(w) = \{s_1, ...s_m\}$. Given a context of w, sense assignment will determine the sense of w by computing the distance of the context to the clusters. Finally, the sense-aware concept vector will be constructed based on the relevance scores of w in the underlying sense. For example, the vectors of “apple” in T1 and T2 are different from each other since they refer to different senses. They only have some relatedness in C5 and C7 where both senses have word occurrences.

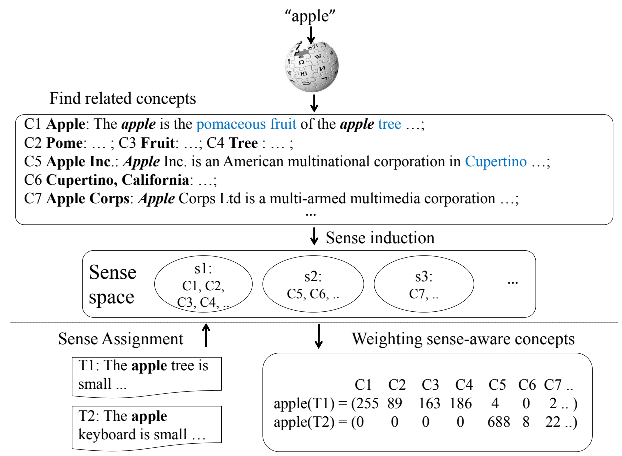

Figure 1: A demonstrative example for *SaSA*

Concept Space

A concept of w should be about w. Or, the Wikipedia article should explicitly mention w (Gabrilovich and Markovitch 2007). However, it is possible that a related article does not mention w, but appears as a linked concept in contexts of w (Hassan and Mihalcea 2011). Thus, to find all related concepts of w, we first find all articles that contain it[2], and then find all linked concepts in contexts of w from those articles. These concepts compose the vector space of w.

To calculate $r_{ij}(w)$, the “relevance” of w to a concept C_{ij}, we define a new TFIDF based measure, namely TFIDF$_s$, to capture the sense-aware relevance. TF is the sum of two parts: number of occurrences of w in C_{ij}, and number of co-occurrences of w and C_{ij} in a sentence in cluster s_i. DF is the number of concepts in the cluster that contains w. When counting the co-occurrences of w and C_{ij}, C_{ij} has to be explicitly marked as a Wikipedia link to the concept C_{ij}. That’s to say, “apple tree” will be counted as one co-occurrence of “apple” and the concept “Tree” i.f.f. “tree” is linked to “http://en.wikipedia.org/wiki/Tree”.

[2]We use Wikipedia API: http://en.wikipedia.org/w/api.php?action =query&list=search&format=json&srsearch=w

One Sense Per Article

As shown in Figure 1, the *one sense per article* assumption made by *SaSA* is not perfect. For example, in the article “Apple Inc.”, among 694 occurrences of “apple”, while most occurrences refer to Apple Inc., there are 4 referring to the fruit apple and 2 referring to “Apple Corps”. However, considering that each Wikipedia article actually focuses on a specific concept, it is still reasonable to believe that the *one sense per article* may hold for most cases. We manually checked all the articles listed in Apple disambiguation page and found that each article has an extremely dominant sense among all occurrences of the word “apple”. Table 1 gives a few examples of sense distribution among four articles. As we can see, each article has a dominant sense. We examined 100 randomly sampled Wikipedia articles and found that 98% of the articles support the assumption. However, considering the two papers “one sense per discourse” (Yarowsky 1993) and “one sense per collocation” (Gale, Church, and Yarowsky 1992a), it would be interesting to see how valid *one sense per article* holds for Wikipedia.

Sense Induction and Assignment

A natural way to find word senses is to use manually created sense inventories such as WordNet (Miller 1995). However, they may miss corpus/domain-specific senses. For example, WordNet provides only two senses for the word

Table 1: Sense distribution examples for the word “apple”

	fruit apple	Apple Inc.	Apple Corps	Apple Bank
Apple	255	0	0	0
Apple Inc.	4	688	2	0
Apple Corps	2	22	193	0
Apple Bank	0	0	0	18

“apple” (food and plant), which is far below the number of senses in Wikipedia. Thus, a more effective way is to automatically discover sense clusters from Wikipedia, possibly by using existing word sense induction techniques plus context clustering (Purandare and Pedersen 2004), where each context is a word vector. However, several problems make this not applicable for our task. First, the computation cost is too high since a word often has a large number of contexts in Wikipedia. For example, “apple” has more than 4 million contexts even if we define our context as large as a paragraph. Second, it is hard to interpret the sense clusters and evaluate the quality of the clustering. In addition, those unlabeled context clusters also add uncertainty and bias for the sense assignment of words in a new context.

By applying *one sense per article*, we can generate sense clusters from Wikipedia articles by hierarchical clustering. Now the question becomes how to decide if two articles or clusters share the sense for a word w. Assume that contexts of w in articles (with the same sense of w) should be similar. We model w’s context in an article using a TF based word vector, which contains two parts: all the Wikipedia concepts (with explicit links) in sentences containing w, and all the words in the dependencies of w from the results of Stanford Parser [3] on the sentences. A cluster’s context is the aggregation of all articles’ contexts of w in the cluster. Suppose the context words of w and the number of their occurrences in concept $C1$ are {t1: 2, t2: 3} and in $C2$ {t2: 2, t3: 3}, then the context of w in the cluster $\{C1, C2\}$ will be {t1: 0.2, t2: 0.5, t3: 0.3}, based on the ratio of each context word’s frequency. We measure two clusters’ context similarity ($ctxSim$) using cosine similarity between their context vectors.

High context similarity could be a good indicator to merge two articles or clusters, if the “sense” of w is well represented by the context vectors. However, there might be cases that it is under represented in an article so that the context vector of the article has a very low similarity to that of the cluster it should belong to. For example, the context vector of “apple” in the article “Gourd” (http://en.wikipedia.org/wiki/Gourd) is {Momordica charantia:1, Momordica Dioica:1, Gac:1, Momordica balsamina:1, Kerala:1, Tamil Nadu:1, balsam: 2}, which has almost a zero similarity score to the context vector of sense cluster {Malus, Apple}. However, we could easily infer that “apple” occurring in “Gourd” would very likely refer to the sense of “apple” in “Malus” or “Apple”, because both share a certain of semantic relatedness at the categorical or topical level, despite the difference of the contexts.

[3] http://nlp.stanford.edu/software/lex-parser.shtml

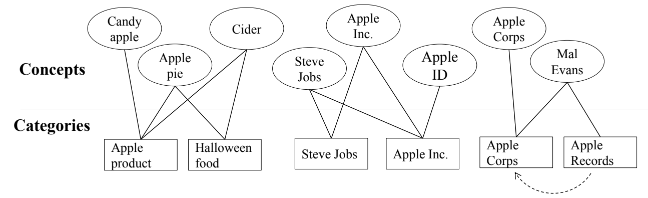

Figure 2: Sense clustering based on Wikipedia categories

How can we model categorical or topical level relatedness between Wikipedia concepts? Notice that categories of Wikipedia articles, which are manually labeled tags, are essentially designed to group pages of similar subjects [4]. For example, the categories of “Malus” include “Eudicot genera”, “Malus”, “Plants and pollinators”, and “Plants with indehiscent fruit”. As expected, the article “Gourd” also has the category “Plants and pollinators”, explicitly showing the connection to “Malus”. However, given a Wikipedia article, not all of its categories are helpful. For example, some categories, such as “All articles lacking sources” or “All categories needing additional references”, have more of a functional role than topical tags. These functional categories are removed based on simple heuristics that are if the number of words is larger than 3 and if it contains one of the words {article, page, error, template, category, people, place, name}. A cluster’s category set consists of all topical categories from the articles in the cluster. Given two clusters s_1 and s_2, and their category sets $G1 = \{x_1, ..., x_p\}$, $G2 = \{y_1, ..., y_q\}$, we define the categorical relatedness between them as a modified Jaccard similairy of $G1$ and $G2$:

$$catSim(s_1, s_2) = \frac{\sum_1^p \sum_1^q rel(x_i, y_j)}{|G1 \cup G2|}$$

where $rel(x_i, y_j)$ is defined as follows:

$$rel(x_i, y_j) = \begin{cases} 1 & x_i = y_j \\ 1/2 & \text{if } x_i \text{ is a subcategory of } y_j \text{ or vice versa} \\ 0 & \text{otherwise} \end{cases}$$

All the concepts C and their categories R form a bipartite graph $G(C, R, E)$, where E denotes the edges between C and R, as shown in Figure 2. Therefore, one may apply bipartite graph clustering algorithms (Zha et al. 2001) on it and regard each cluster as a sense cluster. However, previous clustering algorithms are either designed for document clustering based on document content or for graphs based on graph topology, which cannot take full advantage of the specialty in our task. We define a bipartite clique $q = G_q(C_q, R_q, E_q)$ as a subset of G, where every node pair between C_q and R_q is connected. For example, in Figure 2, “Apple pie”, “Cider”, “Apple product”, and “Halloween food” form a clique. A hidden directed edge between categories denotes one is a subcategory of the other, as shown by the link between “Apple Records” and “Apple Corps”. It is straightforward to regard each clique as a sense cluster candidate. However, our empirical results show that there are always far more clusters than it should be and a lot of cliques contain just single pair of concept-category.

[4] Categories are normally found at the bottom of an article page

Table 2: Pearson (γ), Spearman (ρ) correlations and their harmonic mean (μ) on word-to-word relatedness datasets. The weighted average WA over the three datasets is also reported.

	Pearson				Spearman				Harmonic mean			
	MC30	RG65	WS353	WA	MC30	RG65	WS353	WA	MC30	RG65	WS353	WA
ESA	0.588	−−	0.503	−−	0.727	−−	0.748	−−	0.650	−−	0.602	−−
SSA_s	0.871	0.847	0.622	0.671	0.810	0.830	0.629	0.670	0.839	0.838	0.626	0.671
SSA_c	0.879	0.861	0.590	0.649	0.843	0.833	0.604	0.653	0.861	0.847	0.597	0.651
$SaSA_t$	0.883	0.870	0.721	0.753	0.849	0.841	0.733	0.756	0.866	0.855	0.727	0.754
SaSA	**0.886**	**0.882**	**0.733**	**0.765**	**0.855**	**0.851**	**0.739**	**0.763**	**0.870**	**0.866**	**0.736**	**0.764**

Finally we measure the similarity of two clusters by averaging categorical relatedness and context similarity, i.e. $cluSim = p \cdot ctxSim + (1-p) \cdot catSim$. We empirically set $p = 0.5$. Two clusters will be merged into one if their $cluSim$ is higher than a threshold λ. After the sense clusters are constructed, given w with its context T, we rank the sense clusters based on the cosine similarity of T between the context of the clusters and use the similarity to estimate the probability that the sense of w belongs to the sense cluster s_i, denoted by $p(T, w, s_i)$.

Relatedness

To compute semantic relatedness between two isolated words, we treat all sense clusters equally. Given two words $w1$ and $w2$, each word's concept vector V is computed based on the defined relevance. And the relatedness between the two words is defined as the cosine similarity of their concept vectors. Given $w1$ and $w2$ along with their contexts $T1$ and $T2$, we adopt the relatedness defined by Reisinger and Mooney (2010) on the top K most possible sense clusters of the two words:

$$AvgSimC(w1, w2) =$$
$$\frac{1}{K^2}\sum_{i=1}^{K}\sum_{j=1}^{K} p(T1, w1, s1_i)p(T2, w2, s2_j)d(s1_i, s2_j)$$

where $p(T1, w1, s1_i)$ is the likelihood that $w1$ in $T1$ belongs to the sense cluster $s1_i$ and $d(\cdot,\cdot)$ is a standard distributional similarity measure. Considering top K clusters, instead of a single one, will make *SaSA* more robust. We set $K = 5$ in experiments and use cosine similarity for d.

Given a text fragment $T = (w_1, ..., w_t)$ (assuming one sense in the fragment for each word w_i), its concept vector is defined as the weighted sum of all words' concept vector. For a word w in T, its relevance to a concept $C_{jk} \in s_j$ is defined as

$$r_{jk}(w) = p(T, w, s_j) \cdot \text{TFIDF}_s(w, C_{jk})$$

Two text fragments' relatedness is then defined as the cosine similarity of their concept vectors.

Evaluation

There are two main questions we want to explore in the evaluation. First, can *SaSA* based relatedness measures effectively compute semantic relatedness between words and texts? And second, is the sense clustering technique in *SaSA* effective for sense induction?

Relatedness on Isolated Words

We evaluate *SaSA* on word-to-word relatedness on three standard datasets, using both Pearson correlation γ and Spearman correlation ρ. We follow (Hassan and Mihalcea 2011) by introducing the harmonic mean of the two metrics $\mu = \frac{2\gamma\rho}{\gamma+\rho}$. Rubenstein and Goodenough (Rubenstein and Goodenough 1965) contains 65 word pairs ranging from synonymy pairs to completely unrelated terms, scoring from 0 (not-related) to 4 (perfect synonymy). Miller-Charles (Miller and Charles 1991) is a subset of the Rubenstein and Goodenough dataset, consisting of 30 word pairs, using a scale from 0 to 4. WordSimilarity-353 (Lev Finkelstein and Ruppin 2002) consists of 353 word pairs annotated on a scale from 0 (unrelated) to 10 (very closely related or identical). It includes verbs, adjectives, names and technical terms, where most of them have multiple senses, therefore posing more difficulty for relatedness metrics.

We compare *SaSA* with ESA (Gabrilovich and Markovitch 2007) and SSA (Hassan and Mihalcea 2011) that have shown better performance than other methods in the literature on the three datasets. The correlation results are shown in Table 2, where SSA_s and SSA_c denote SSA using second order co-occurrence point mutual information (Islam and Inkpen 2006) and SSA using cosine respectively (Hassan and Mihalcea 2011). $SaSA_t$ is a modified *SaSA* that uses traditional TFIDF for relevance, whose concept space can be regarded as a "union" of ESA and SSA. It outperforms ESA and SSA in both Pearson and Spearman correlation, indicating it models a more comprehensive concept space for a word. *SaSA* gains slight further improvement over $SaSA_t$, showing the effectiveness of the new relevance.

Relatedness on Words in Sentential Contexts

While isolated word-to-word relatedness can only be measured in the sense-unaware style, relatedness on words in contexts enables *SaSA* to do sense assignment based on a word's context. We compare our model with existing methods on the sentential contexts dataset (Huang et al. 2012), which contains a total of 2,003 word pairs, their sentential contexts, the 10 individual human ratings in [0,10], as well as their averages. Table 3 shows different models' results on the dataset based on Spearman (ρ) correlation.[5] Pruned tfidf-M represents Huang et al.'s implementation of Reisinger and Mooney (2010). Huang et al. 2012 refers to

[5] Pearson (γ) was not reported in Huang et al.'s paper.

Table 3: Spearman (ρ) correlation on the sentential context dataset (Huang et al. 2012)

Model	ρ
ESA	0.518
SSA	0.509
Pruned tfidf-M	0.605
Huang et al. 2012	0.657
$SaSA_1$	**0.662**
$SaSA_K$	**0.664**

Table 4: *V-Measure* and *F-Score* for word sense induction on 10 words

words	*V-M*	*F-S*	words	*V-M*	*F-S*
book	0.165	0.634	doctor	0.153	0.660
dog	0.153	0.652	company	0.155	0.654
tiger	0.149	0.678	stock	0.147	0.633
plane	0.171	0.723	bank	0.148	0.682
train	0.174	0.758	king	0.166	0.693

their best results. As shown by Table 3, our *SaSA* models consistently outperform single prototype models ESA and SSA, and multi-prototype models of both Reisinger and Mooney (2010) and Huang et al. (2012). $SaSA_1$ uses only the closest sense cluster to build the concept vector while $SaSA_K$ considers the top $K(=5)$ clusters. The results clearly show the advantage of $SaSA$ over both Wikipedia based single prototype models and free text based multi-prototype models. For this relatedness on words in sentential contexts task, we also did sensitive study for the parameter K and the threshold λ. We found the performance keeps improving as K increases when $K <= 5$ and then stays stable after that. We also found that $\lambda \in [0.12, 0.18]$ gives the best results.

Relatedness on texts

To measure the relatedness on texts, we also use three standard datasets that have been used in the past. Lee50 (Lee, Pincombe, and Welsh 2005) consists of 50 documents collected from the Australian Broadcasting Corporation's news mail service. Every document pair is scored by ten annotators, resulting in 2,500 annotated document pairs with their similarity scores. The evaluations are carried out on only 1225 document pairs after ignoring duplicates. Li30 (Li et al. 2006) is a sentence pair similarity dataset constructed using the definition pairs of Rubenstein and Goodenough wordpairs (Rubenstein and Goodenough 1965). AG400 (Mohler and Mihalcea 2009) consists of 630 student answers along with the corresponding questions and correct answers. Each student answer was graded by two judges scaling from 0 to 5, where 0 means completely wrong and 5 indicates perfect. We followed previous work (Hassan and Mihalcea 2011) and randomly eliminated 230 of the highest grade answers to produce more normally distributed scores.

The comparison results of *SaSA* with the baselines on the three datasets are shown in Table 5. It clearly demonstrates that *SaSA* outperforms all the other baselines in terms of all correlations. Besides, it is interesting to note that $SaSA_1$ has better performance than $SaSA_K$ in Li30 and Lee50, but worse results in AG400. The reason could be that the former two datasets are constructed using more formal resource, such as definitions or news, whose textual similarity to Wikipedia concepts is much higher than the AG400 dataset based on student/teacher QAs.

Sense Induction

Performance on relatedness implies a high quality of sense clusters generated by $SaSA$. To demonstrate the results in a more explicit way, we select 10 words and manually judge the clustering results of the top 200 concepts returned by the Wikipedia API for each word. The evaluation metrics are *V-Measure* (Rosenberg and Hirschberg 2007) and *paired F-Score* (Artiles, Amig, and Gonzalo 2009). *V-measure* assesses the quality of a cluster by measuring its homogeneity and completeness. Homogeneity measures the degree that each cluster consists of points primarily belonging to a single GS (golden standard) class, while completeness measures the degree that each GS class consists of points primarily assigned to a single cluster. Similar to traditional *F-Score*, *paired F-Score* is defined based on the precision and recall of instance pairs, where precision measures the fraction of GS instance pairs in a cluster while recall measures the ratio of GS instance pairs in a cluster to the total number of instance pairs in the GS cluster.

The average *V-Measure* and *paired F-Score* over the 10 words are 0.158 and 0.677 respectively, which are as high as the best reported results in sense induction (Manandhar et al. 2010). Detailed results of each word are in Table 4, showing the consistent performance of *SaSA* on all the words. Exemplary concepts in the top 3 largest clusters of "apple", "jaguar" and "stock" are shown in Table 6, where we can find that each cluster has a reasonable sense.

In general, larger λ increases the homogeneity but decreases the completeness of a sense cluster. If a word itself is a Wikipedia title and its context information is rich, setting a smaller K could give better representations. On a Red Hat Linux Server(5.7) with 2.35GHz Intel(R) Xeon(R) 4 processor and 23GB of RAM, we can build a sense-aware profile for a word from the datasets within 2 minutes using the Wikipedia API, which is comparable to ESA and SSA.

Related Work

Most existing work on word embedding ignore words of multiple senses and build a single vector representation, with a few exceptions such as Reisinger and Mooney (2010) and Huang et al. (2012). They both assume a fix predefined number of clusters for all words and apply text based clustering to do sense induction. We take advantage of Wikipedia to generate more accurate sense clusters.

Semantic relatedness measures can be roughly grouped into two main categories: knowledge-based and corpus-based. Knowledge-based measures such as (Lesk 1986; Resnik 1995; Hirst and St-Onge 1998; Leacock and Chodorow 1998), leverage information extracted from manually constructed taxonomies such as Wordnet (Miller 1995; Agirre et al. 2009; Pilehvar, Jurgens, and Navigli 2013)

Table 5: Pearson (γ), Spearman (ρ) correlations and their harmonic mean (μ) on text-to-text relatedness datasets. The weighted average WA over the three datasets is also reported.

	Pearson				Spearman				Harmonic mean			
	Li30	Lee50	AG400	WA	Li30	Lee50	AG400	WA	Li30	Lee50	AG400	WA
ESA	0.838	0.696	0.365	0.622	0.863	0.463	0.318	0.433	0.851	0.556	0.340	0.512
SSA_s	0.881	0.684	0.567	0.660	0.878	0.480	0.495	0.491	0.880	0.564	0.529	0.561
SSA_c	0.868	0.684	0.559	0.658	0.870	0.488	0.478	0.492	0.869	0.569	0.515	0.562
$SaSA_1$	**0.895**	**0.732**	0.576	**0.697**	**0.902**	**0.626**	0.518	**0.604**	**0.898**	**0.675**	0.545	**0.648**
$SaSA_K$	0.887	0.715	**0.592**	0.688	0.893	0.609	**0.526**	0.594	0.890	0.658	**0.557**	0.644

Table 6: Examples of top 3 sense clusters discovered by *SaSA*

words	sense cluster 1	sense cluster 2	sense cluster 3
apple	Apple Inc., Steve Jobs, Macintosh, IPod, IPad, Apple TV, IPhone, IOS, ITunes, Apple A6X, Apple I, ...	Apple, Malus, Cider, Apple butter, Candy apple, Apple cake, Apple crisp, Apple cider, Apple source, ...	Apple Corps, Apple scruffs, Apple Boutique, Apple Records, The Beatles, Mal Evans, ...
jaguar	Jaguar Cars, Jaguar Racing, Tom Walkinshaw Racing, Jaguar R1, Jaguar XJ, Jaguar XK, Jaguar S-Type, ...	Jaguar, Black Jaguar, European jaguar, Paseo del Jaguar, Panthera, Big cat, Leopard, ...	Jacksonville Jaguars, History of the Jacksonville Jaguars, ...
stock	Stock, Stock market, Common stock, Stock exchange, Penny stock, Stock market index, Shareholder, ...	Inventory, Stock and flow, Stock management, Stock keeping unit, Safety stock, Stock control, ...	Rolling stock, London Underground rolling stock, London Underground D78 Stock, ...

and Wiktionary (Zesch, Müller, and Gurevych 2008). While they show potential in measuring semantic relatedness, the strong dependence on static, expensive, manually constructed taxonomies often limits their applicability. Moreover, they are not readily portable across languages, since their application to a new language requires the availability of a lexical resource in that language.

Corpus-based measures model semantics of text using probabilistic approaches, by leveraging contextual information of words in the corpus, based on the *distributional hypothesis* (Harris 1981). Most of this work, such as Pointwise Mutual Information (PMI) (Church and Hanks 1990), distributional similarity (Lin 1998), PMI-IR (Turney 2001), Second Order PMI (Islam and Inkpen 2006), WikiRelate!(Strube and Ponzetto 2006), builds a semantic profile for a word using a word vector space based on word co-occurrence, while more recent works, such as LSA (Landauer et al. 1991), ESA (Gabrilovich and Markovitch 2007), WikiWalk (Yeh et al. 2009), SSA (Hassan and Mihalcea 2011), and TSA (Radinsky et al. 2011), employ a concept/document-based approach to build a concept space for a word, with the semantic profile expressed in terms of the explicit (ESA, SSA, and TSA) or implicit (LSA) concepts. The explicit concepts are defined as Wikipedia articles that relate to a word, while the implicit concepts are derived from term-document association matrix using singular value decomposition. Though concept-based methods can deal with the problems of word-space methods such as word ambiguousness and vocabulary mismatch, they are still sense-unaware. There is also a growing interest in building word embeddings using neural networks from free text corpus (Huang et al. 2012; Mikolov et al. 2013) and from knowledge bases (Bordes et al. 2011). What differs our *SaSA* from them is that *SaSA* builds a sense-aware concept-based semantic profile for a word under a certain sense, which we argue addresses the word sense ambiguousness problem in a more fundamental way.

It's important to note that Wikipedia has been widely studied as a knowledge base for word sense induction and disambiguation (Mihalcea 2007; Ponzetto and Navigli 2010), entity disambiguation (Mihalcea and Csomai 2007; Cucerzan 2007), and term extraction (Wu and Giles 2013; Wu et al. 2013). Besides, other semantic resources such as BabelNet have been used for similar studies (Navigli and Ponzetto 2012; Moro, Raganato, and Navigli 2014). However, the main focus of *SaSA* is to provide a semantic modeling approach that can better capture semantic relatedness of texts, not to address the tasks of word sense disambiguation or name entity disambiguation.

Conclusion and Future Work

We present sense-aware semantic analysis (*SaSA*), a distributional semantic modeling method that models a word in the sense level, by conducting sense induction from the related Wikipedia concepts. Evaluations on various semantic relatedness measurement tasks demonstrate its effectiveness. It significantly outperforms the best reported methods in both single prototype and multi-prototype models.

Although Wikipedia is the largest encyclopedia knowledge base online with wide knowledge coverage, it is still possible that some word senses could be under represented or even absent. They could be words in other languages not well covered by Wikipedia, or newly created words or existing words with new senses that have emerged from news or the social media. Promising future work would be to build sense-ware representation models for words using other corpora or knowledge bases, e.g., news, tweets, or structural knowledge bases such as Freebase. Another direction would be devising better algorithms and incorporating other sense inventories to improve sense induction.

Acknowledgments

We gratefully acknowledge partial support from the NSF.

References

Agirre, E.; Alfonseca, E.; Hall, K.; Kravalova, J.; Paşca, M.; and Soroa, A. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *NAACL*, 19–27.

Artiles, J.; Amig, E.; and Gonzalo, J. 2009. The role of named entities in web people search. In *EMNLP*, 534–542.

Bordes, A.; Weston, J.; Collobert, R.; and Bengio, Y. 2011. Learning structured embeddings of knowledge bases. In *AAAI*, 301–306.

Church, K. W., and Hanks, P. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics* 1(16):22–29.

Cucerzan, S. 2007. Large-scale named entity disambiguation based on wikipedia data. In *EMNLP-CoNLL*, 708–716.

Deerwester, S.; Dumais, S. T.; Furnas, G. W.; Landauer, T. K.; and Harshman, R. 1990. Indexing by latent semantic analysis. *JASIS* 41(6):391–407.

Gabrilovich, E., and Markovitch, S. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, 1606–1611.

Gale, W. A.; Church, K. W.; and Yarowsky, D. 1992a. One sense per discourse. In *Workshop on Speech and Natural Language*, 233–237.

Gale, W. A.; Church, K. W.; and Yarowsky, D. 1992b. A method for disambiguating word senses in a large corpus. *Computers and the Humanities* 26(5-6):415–439.

Harris, Z. 1981. Distributional structure. In *Papers on Syntax*, volume 14. Springer Netherlands. 3–22.

Hassan, S., and Mihalcea, R. 2011. Semantic relatedness using salient semantic analysis. In *AAAI*, 884–889.

Hirst, G., and St-Onge, D. 1998. Lexical chains as representations of context for the detection and correction of malapropisms. In *WordNet: An Electronic Lexical Database*, 305–332.

Huang, E. H.; Socher, R.; Manning, C. D.; and Ng, A. Y. 2012. Improving word representations via global context and multiple word prototypes. In *ACL*, 873–882.

Islam, A., and Inkpen, D. 2006. Second order co-occurrence pmi for determining the semantic similarity of words. In *LREC*, 1033–1038.

Landauer, T. K.; Laham, D.; Rehder, B.; and Schreiner, M. E. 1991. How well can passage meaning be derived without using word order:a comparison of latent semantic analysis and humans. In *CogSci*, 412–417.

Leacock, C., and Chodorow, M. 1998. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database* 49(2):265–283.

Lee, M. D.; Pincombe, B.; and Welsh, M. B. 2005. An empirical evaluation of models of text document similarity. *Cognitive Science* 1254–1259.

Lesk, M. 1986. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *SIGDOC*, 24–26.

Lev Finkelstein, Evgeniy Gabrilovich, Y. M. E. R. Z. S. G. W., and Ruppin, E. 2002. Placing search in context: The concept revisited. *TOIS* 1(20):116–131.

Li, Y.; McLean, D.; Bandar, Z.; O'Shea, J.; and Crockett, K. A. 2006. Sentence similarity based on semantic nets and corpus statistics. *TKDE* 18(8):1138–1150.

Lin, D. 1998. An information-theoretic definition of similarity. In *ICML*, 296–304.

Manandhar, S.; Klapaftis, I. P.; Dligach, D.; and Pradhan, S. S. 2010. Semeval-2010 task 14: Word sense induction & disambiguation. In *SemEval*, 63–68.

Mihalcea, R., and Csomai, A. 2007. Wikify!: Linking documents to encyclopedic knowledge. In *CIKM*, 233–242.

Mihalcea, R. 2007. Using wikipedia for automatic word sense disambiguation. In *NAACL*, 196–203.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, 3111–3119.

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

Miller, G. A. 1995. Wordnet: A lexical database for english. *Communications of the ACM* 38(11):39–41.

Mohler, M., and Mihalcea, R. 2009. Text-to-text semantic similarity for automatic short answer grading. In *EACL*, 567–575.

Moro, A.; Raganato, A.; and Navigli, R. 2014. Entity linking meets word sense disambiguation: A unified approach. *TACL* 2:231–244.

Navigli, R., and Ponzetto, S. P. 2012. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence* 193:217–250.

Pantel, P., and Lin, D. 2002. Discovering word senses from text. In *KDD*, 613–619.

Pilehvar, M. T.; Jurgens, D.; and Navigli, R. 2013. Align, disambiguate and walk: A unified approach for measuring semantic similarity. In *ACL*, 1341–1351.

Ponzetto, S. P., and Navigli, R. 2010. Knowledge-rich word sense disambiguation rivaling supervised systems. In *ACL*, 1522–1531.

Purandare, A., and Pedersen, T. 2004. Word sense discrimination by clustering contexts in vector and similarity spaces. In *CoNLL*, 41–48.

Radinsky, K.; Agichtein, E.; Gabrilovich, E.; and Markovitch, S. 2011. A word at a time: Computing word relatedness using temporal semantic analysis. In *WWW*, 337–346.

Reisinger, J., and Mooney, R. J. 2010. Multi-prototype vector-space models of word meaning. In *NAACL*, 109–117.

Resnik, P. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI*, 448–453.

Rosenberg, A., and Hirschberg, J. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *EMNLP-CoNLL*, 410–420.

Rubenstein, H., and Goodenough, J. B. 1965. Contextual correlates of synonymy. *Commun. ACM* 8(10):627–633.

Schütze, H. 1998. Automatic word sense discrimination. *Comput. Linguist.* 24(1):97–123.

Strube, M., and Ponzetto, S. P. 2006. Wikirelate! computing semantic relatedness using wikipedia. In *AAAI*, 1419–1424.

Turney, P. 2001. Mining the web for synonyms: Pmi-ir versus lsa on toefl. In *ECML*, 491–502.

Wu, Z., and Giles, C. L. 2013. Measuring term informativeness in context. In *NAACL*, 259–269.

Wu, Z.; Li, Z.; Mitra, P.; and Giles, C. L. 2013. Can back-of-the-book indexes be automatically created? In *CIKM*, 1745–1750.

Yarowsky, D. 1993. One sense per collocation. In *Workshop on Human Language Technology*, 266–271.

Yeh, E.; Ramage, D.; Manning, C. D.; Agirre, E.; and Soroa, A. 2009. Wikiwalk: random walks on wikipedia for semantic relatedness. In *Workshop on Graph-based Methods for NLP*, 41–49.

Zesch, T.; Müller, C.; and Gurevych, I. 2008. Using wiktionary for computing semantic relatedness. In *AAAI*, 861–866.

Zha, H.; He, X.; Ding, C.; Simon, H.; and Gu, M. 2001. Bipartite graph partitioning and data clustering. In *CIKM*, 25–32.