Unsupervised Translation Sense Clustering

Mohit Bansal[*]
UC Berkeley
mbansal@cs.berkeley.edu

John DeNero
Google
denero@google.com

Dekang Lin
Google
lindek@google.com

Abstract

We propose an unsupervised method for clustering the translations of a word, such that the translations in each cluster share a common semantic sense. Words are assigned to clusters based on their usage distribution in large monolingual and parallel corpora using the soft K-Means algorithm. In addition to describing our approach, we formalize the task of translation sense clustering and describe a procedure that leverages WordNet for evaluation. By comparing our induced clusters to reference clusters generated from WordNet, we demonstrate that our method effectively identifies sense-based translation clusters and benefits from both monolingual and parallel corpora. Finally, we describe a method for annotating clusters with usage examples.

1 Introduction

The ability to learn a bilingual lexicon from a parallel corpus was an early and influential area of success for statistical modeling techniques in natural language processing. Probabilistic word alignment models can induce bilexical distributions over target-language translations of source-language words (Brown et al., 1993). However, word-to-word correspondences do not capture the full structure of a bilingual lexicon. Consider the example bilingual dictionary entry in Figure 1; in addition to enumerating the translations of a word, the dictionary author has grouped those translations into three sense clusters. Inducing such a clustering would prove useful in generating bilingual dictionaries automatically or building tools to assist bilingual lexicographers.

Colocar [co·lo·car´], *va.* 1. To arrange, to put in due place or order. 2. To place, to put in any place, rank condition or office, to provide a place or employment. 3. To collocate, to locate, to lay.

Figure 1: This excerpt from a bilingual dictionary groups English translations of the polysemous Spanish word *colocar* into three clusters that correspond to different word senses (Velázquez de la Cadena et al., 1965).

This paper formalizes the task of clustering a set of translations by sense, as might appear in a published bilingual dictionary, and proposes an unsupervised method for inducing such clusters. We also show how to add usage examples for the translation sense clusters, hence providing complete structure to a bilingual dictionary.

The input to this task is a set of source words and a set of target translations for each source word. Our proposed method clusters these translations in two steps. First, we induce a global clustering of the entire target vocabulary using the soft K-Means algorithm, which identifies groups of words that appear in similar contexts (in a monolingual corpus) and are translated in similar ways (in a parallel corpus). Second, we derive clusters over the translations of each source word by projecting the global clusters.

We evaluate these clusters by comparing them to reference clusters with the overlapping BCubed metric (Amigo et al., 2009). We propose a clustering criterion that allows us to derive reference clusters from the synonym groups of WordNet® (Miller, 1995).[1]

Our experiments using Spanish-English and Japanese-English datasets demonstrate that the automatically generated clusters produced by our method are substantially more similar to the

[*] Author was a summer intern with Google Research while conducting this research project.

[1] WordNet is used only for evaluation; our sense clustering method is fully unsupervised and language-independent.

Sense cluster	**WordNet sense description**	**Usage example**
collocate	*group or chunk together in a certain order or place side by side*	colocar juntas todas los libros *collocate all the books*
invest, place, put	*make an investment*	capitales para colocar *capital to invest*
locate, place	*assign a location to*	colocar el número de serie *locate the serial number*
place, position, put	*put into a certain place or abstract location*	colocar en un lugar *put in a place*

Figure 2: Correct sense clusters for the translations of Spanish verb s = *colocar*, assuming that it has translation set T_s = {*collocate, invest, locate, place, position, put*}. Only the sense clusters are outputs of the translation sense clustering task; the additional columns are presented for clarity.

WordNet-based reference clusters than naive baselines. Moreover, we show that bilingual features collected from parallel corpora improve clustering accuracy over monolingual distributional similarity features alone.

Finally, we present a method for annotating clusters with usage examples, which enrich our automatically generated bilingual dictionary entries.

2 Task Description

We consider a three-step pipeline for generating structured bilingual dictionary entries automatically.

(1) The first step is to identify a set of high-quality target-side translations for source lexical items. In our experiments, we ask bilingual human annotators to create these translation sets.[2] We restrict our present study to word-level translations, disallowing multi-word phrases, in order to leverage existing lexical resources for evaluation.

(2) The second step is to cluster translations of each word according to common word senses. This clustering task is the primary focus of the paper, and we formalize it in this section.

(3) The final step annotates clusters with usage examples to enrich the structure of the output. Section 7 describes a method of identifying cluster-specific usage examples.

In the task of *translation sense clustering*, the second step, we assume a fixed set of source lexical items of interest S, each with a single part of speech[3], and for each $s \in S$ a set T_s of target translations. Moreover, we assume that each target word $t \in T_s$ has a set of senses in common with s. These senses may also be shared among different target words. That is, each target word may have multiple senses and each sense may be expressed by multiple words.

Given a translation set T_s, we define a cluster $G \subseteq T_s$ to be a *correct sense cluster* if it is both *coherent* and *complete*.

- A sense cluster G is **coherent** if and only if there exists some sense B shared by all of the target words in G.
- A sense cluster G is **complete** if and only if, for every sense B shared by all words in G, there is no other word in T_s but not in G that also shares that sense.

The full set of correct clusters for a set of translations consists of all sense clusters that are both coherent and complete.

The example translation set for the Spanish word *colocar* in Figure 2 is shown with four correct sense clusters. For descriptive purposes, these clusters are annotated by WordNet senses and bilingual usage examples. However, the task we have defined does not require the WordNet sense or usage example to be identified: we must only produce the correct sense clusters within a set of translations. In fact, a cluster may correspond to more than one sense.

Our definition of correct sense clusters has several appealing properties. First, we do not attempt to enumerate all senses of the source word. Sense

[2] We do not use automatically extracted translation sets in our experiments, in order to isolate the clustering task on clean input.

[3] A noun and verb that share the same word form would constitute two different source lexical items.

Notation

T_s : The set of target-language translations (given)
$\mathcal{D}_t$: The set of synsets in which t appears (given)
C : A synset; a set of target-language words
B : A source-specific synset; a subset of T_s
$\mathcal{B}$: A set of source-specific synsets
$\mathcal{G}$: A set of correct sense clusters for T_s

The Cluster Projection Algorithm:

```
$\mathcal{B} \leftarrow \left\{ C \cap T_s \,:\, C \in \bigcup_{t \in T_s} \mathcal{D}_t \right\}$
$\mathcal{G} \leftarrow \emptyset$
**for** $B \in \mathcal{B}$ **do**
    **if** $\nexists B' \in \mathcal{B}$ such that $B \subset B'$ **then**
        add $B$ to $\mathcal{G}$
**return** $\mathcal{G}$
```

Figure 3: The Cluster Projection (CP) algorithm projects language-level synsets (C) to source-specific synsets (B) and then filters the set of synsets for redundant subsets to produce the complete set of source-specific synsets that are both coherent and complete ($\mathcal{G}$).

Words	Synsets	Sense Clusters
collocate	collocate collocate, lump, chunk	collocate
invest	invest, put, commit, place invest, clothe, adorn invest, vest, enthrone …	invest, place, put locate, place
locate	locate, turn up situate, locate locate, place, site …	place, position, put
place	put, set, place, pose, position, lay rate, rank, range, order, grade, place locate, place, site invest, put, commit, place …	
position	position put, set, place, pose, position, lay	
put	put, set, place, pose, position, lay put frame, redact, cast, put, couch invest, put, commit, place …	

Figure 4: An example of cluster projection on WordNet, for the Spanish source word *colocar*. We show the target translation words to be clustered, their WordNet synsets (with words not in the translation set grayed out), and the final set of correct sense clusters.

distinctions are only made when they affect cross-lingual lexical choice. If a source word has many fine-grained senses but translates in the same way regardless of the sense intended, then there is only one correct sense cluster for that translation.

Second, no correct sense cluster can be a superset of another, because the subset would violate the completeness condition. This criterion encourages larger clusters that are easier to interpret, as their unifying senses can be identified as the intersection of senses of the translations in the cluster.

Third, the correct clusters need not form a partition of the input translations. It is common in published bilingual dictionaries for a translation to appear in multiple sense clusters. In our example, the polysemous English verbs *place* and *put* appear in multiple clusters.

3 Generating Reference Clusters

To construct a reference set for the translation sense clustering task, we first collected English translations of Spanish and Japanese nouns, verbs, and adverbs. Translation sets were curated by human annotators to keep only high-quality single-word translations.

Rather than gathering reference clusters via an additional annotation effort, we leverage WordNet, a large database of English lexical semantics (Miller, 1995). WordNet groups words into sets of cognitive synonyms called *synsets*, each expressing a distinct concept. We use WordNet version 2.1, which has wide coverage of nouns, verbs, and adverbs, but sparser coverage of adjectives and prepositions.[4]

Reference clusters for the set of translations T_s of some source word s are generated algorithmically from WordNet synsets via the *Cluster Projection* (CP) algorithm defined in Figure 3. An input to the CP algorithm is the translation set T_s of some source word s. Also, each translation $t \in T_s$ belongs to some set of synsets $\mathcal{D}_t$, where each synset $C \in D_t$ contains target-language words that may or may not be translations of s. First, the CP algorithm constructs a source-specific synset B for each C, which contains only translations of s. Second, it identifies all correct sense clusters $\mathcal{G}$ that are both *coherent* and *complete* with respect to the source-specific senses $\mathcal{B}$. A sense cluster must correspond to some synset $B \in \mathcal{B}$ to be coherent, and it must

[4] WordNet version 2.1 is almost identical to version 3.0, for Unix-like systems, as described in http://wordnetcode.princeton.edu/3.0/CHANGES. The latest version 3.1 is not yet available for download.

not have a proper superset in $\mathcal{B}$ to be complete.[5]

Figure 4 illustrates the CP algorithm for the translations of the Spanish source word *colocar* that appear in our input dataset.

4 Clustering with K-Means

In this section, we describe an unsupervised method for inducing translation sense clusters from the usage statistics of words in large monolingual and parallel corpora. Our method is language independent.

4.1 Distributed Soft K-Means Clustering

As a first step, we cluster all words in the target-language vocabulary in a way that relates words that have similar distributional features. Several methods exist for this task, such as the K-Means algorithm (MacQueen, 1967), the Brown algorithm (Brown et al., 1992) and the exchange algorithm (Kneser and Ney, 1993; Martin et al., 1998; Uszkoreit and Brants, 2008). We use a distributed implementation of the "soft" K-Means clustering algorithm described in Lin and Wu (2009). Given a feature vector for each element (a word type) and the number of desired clusters K, the K-Means algorithm proceeds as follows:

```
1. Select $K$ elements as the initial centroids for $K$ clusters.
repeat
    2. Assign each element to the top $M$ clusters with the nearest centroid, according to a similarity function in feature space.
    3. Recompute each cluster's centroid by averaging the feature vectors of the elements in that cluster.
until convergence
```

4.2 Monolingual Features

Following Lin and Wu (2009), each word to be clustered is represented as a feature vector describing the distributional context of that word. In our setup, the context of a word w consists of the words immediately to the left and right of w. The context feature vector of w is constructed by first aggregating the frequency counts of each word f in the context of each w. We then compute point-wise mutual information (PMI) features from the frequency counts:

$$\text{PMI}(w, f) = \log \frac{\text{c}(w, f)}{\text{c}(w)\text{c}(f)}$$

where w is a word, f is a neighboring word, and $c(\cdot)$ is the count of a word or word pair in the corpus.[6] A feature vector for w contains a PMI feature for each word type f (with relative position left or right) for all words that appears a sufficient number of times as a neighbor of w. The similarity of two feature vectors is the cosine of the angle between the vectors. We follow Lin and Wu (2009) in applying various thresholds during K-Means, such as a frequency threshold for the initial vocabulary, a total-count threshold for the feature vectors, and a threshold for PMI scores.

4.3 Bilingual Features

In addition to the features described in Lin and Wu (2009), we introduce features from a bilingual parallel corpus that encode *reverse-translation* information from the source-language (Spanish or Japanese in our experiments). We have two types of bilingual features: unigram features capture source-side reverse-translations of w, while bigram features capture both the reverse-translations and source-side neighboring context words to the left and right. Features are expressed again as PMI computed from frequency counts of aligned phrase pairs in a parallel corpus. For example, one unigram feature for *place* would be the PMI computed from the number of times that *place* was in the target side of a phrase pair whose source side was the unigram *lugar*. Similarly, a bigram feature for *place* would be the PMI computed from the number of times that *place* was in the target side of a phrase pair whose source side was the bigram *lugar de*. These features characterize the way in which a word is translated, an indication of its meaning.

[5] One possible shortcoming of our approach to constructing reference sets for translation sense clustering is that a cluster may correspond to a sense that is not shared by the original source word used to generate the translation set. All translations must share some sense with the source word, but they may not share all senses with the source word. It is possible that two translations are synonymous in a sense that is not shared by the source. However, we did not observe this problem in practice.

[6] PMI is typically defined in terms of probabilities, but has proven effective previously when defined in terms of counts.

4.4 Predicting Translation Clusters

As a result of soft K-Means clustering, each word in the target-language vocabulary is assigned to a list of up to M clusters. To predict the sense clusters for a set of translations of a source word, we apply the CP algorithm (Figure 3), treating the K-Means clusters as synsets ($\mathcal{D}_t$).

5 Related Work

To our knowledge, the translation sense clustering task has not been explored previously. However, much prior work has explored the related task of monolingual word and phrase clustering. Uszkoreit and Brants (2008) uses an exchange algorithm to cluster words in a language model, Lin and Wu (2009) uses distributed K-Means to cluster phrases for various discriminative classification tasks, Vlachos et al. (2009) uses Dirichlet Process Mixture Models for verb clustering, and Sun and Korhonen (2011) uses a hierarchical Levin-style clustering to cluster verbs.

Previous word sense induction work (Diab and Resnik, 2002; Kaji, 2003; Ng et al., 2003; Tufis et al., 2004; Apidianaki, 2009) relates to our work in that these approaches discover word senses automatically through clustering, even using multilingual parallel corpora. However, our task of clustering multiple words produces a different type of output from the standard word sense induction task of clustering in-context uses of a single word. The underlying notion of "sense" is shared across these tasks, but the way in which we use and evaluate induced senses is novel.

6 Experiments

The purpose of our experiments is to assess whether our unsupervised soft K-Means clustering method can effectively recover the reference sense clusters derived from WordNet.

6.1 Datasets

We conduct experiments using two bilingual datasets: Spanish-to-English (S→E) and Japanese-to-English (J→E). Table 1 shows, for each dataset, the number of source words and the total number of target words in their translation sets. The datasets are limited in size because we solicited human annotators to filter the set of translations for each source word. The S→E dataset has 52 source-words with a part-of-speech-tag distribution of 38 nouns, 10 verbs and 4 adverbs. The J→E dataset has 369 source-words with 319 nouns, 38 verbs and 12 adverbs. We included only these parts of speech because WordNet version 2.1 has adequate coverage for them. Most source words have 3 to 5 translations each.

Dataset	No. of src-words	Total no. of tgt-words
S→E	52	230
J→E	369	1639

Table 1: Sizes of the Spanish-to-English (S→E) and Japanese-to-English (J→E) datasets.

Monolingual features for K-Means clustering were computed from an English corpus of Web documents with 700 billion tokens of text. Bilingual features were computed from 0.78 (S→E) and 1.04 (J→E) billion tokens of parallel text, primarily extracted from the Web using automated parallel document identification (Uszkoreit et al., 2010). Word alignments were induced from the HMM-based alignment model (Vogel et al., 1996), initialized with the bilexical parameters of IBM Model 1 (Brown et al., 1993). Both models were trained using 2 iterations of the expectation maximization algorithm. Phrase pairs were extracted from aligned sentence pairs in the same manner used in phrase-based machine translation (Koehn et al., 2003).

6.2 Clustering Evaluation Metrics

The quality of text clustering algorithms can be evaluated using a wide set of metrics. For evaluation by set matching, the popular measures are Purity (Zhao and Karypis, 2001) and Inverse Purity and their harmonic mean (F measure, see Van Rijsbergen (1974)). For evaluation by counting pairs, the popular metrics are the Rand Statistic and Jaccard Coefficient (Halkidi et al., 2001; Meila, 2003).

Metrics based on entropy include Cluster Entropy (Steinbach et al., 2000), Class Entropy (Bakus et al., 2002), VI-measure (Meila, 2003), Q_0 (Dom, 2001), V-measure (Rosenberg and Hirschberg, 2007) and Mutual Information (Xu et al., 2003). Lastly, there exist the BCubed metrics (Bagga and Baldwin, 1998), a family of metrics that decompose the clustering evaluation by estimating precision and recall for each item in the distribution.

Amigo et al. (2009) compares the various clustering metrics mentioned above and their properties. They define four formal but intuitive constraints on such metrics that explain which aspects of clustering quality are captured by the different metric families. Their analysis shows that of the wide range of metrics, only BCubed satisfies those constraints. After defining each constraint below, we briefly describe its relevance to the translation sense clustering task.

Homogeneity: In a cluster, we should not mix items belonging to different categories.

Relevance: All words in a proposed cluster should share some common WordNet sense.

Completeness: Items belonging to the same category should be grouped in the same cluster.

Relevance: All words that share some common WordNet sense should appear in the same cluster.

Rag Bag: Introducing disorder into a disordered cluster is less harmful than introducing disorder into a clean cluster.

Relevance: We prefer to maximize the number of error-free clusters, because these are most easily interpreted and therefore most useful.

Cluster Size vs. Quantity: A small error in a big cluster is preferable to a large number of small errors in small clusters.

Relevance: We prefer to minimize the total number of erroneous clusters in a dictionary.

Amigo et al. (2009) also show that BCubed extends cleanly to settings with overlapping clusters, where an element can simultaneously belong to more than one cluster. For these reasons, we focus on BCubed for cluster similarity evaluation.[7]

The BCubed metric for scoring overlapping clusters is computed from the pair-wise precision and recall between pairs of items:

$$\mathrm{P}(e,e') = \frac{\min(|C(e)\cap C(e')|,\, |L(e)\cap L(e')|)}{|C(e)\cap C(e')|}$$
$$\mathrm{R}(e,e') = \frac{\min(|C(e)\cap C(e')|,\, |L(e)\cap L(e')|)}{|L(e)\cap L(e')|}$$

where e and e' are two items, $L(e)$ is the set of reference clusters for e and $C(e)$ is the set of predicted clusters for e (i.e., clusters to which e belongs). Note that $\mathrm{P}(e,e')$ is defined only when e and e' share some predicted cluster, and $\mathrm{R}(e,e')$ when e and e' share some reference cluster.

The BCubed precision associated to one item is its averaged pair-wise precision over other items sharing some of its predicted clusters, and likewise for recall[8]; and the *overall* BCubed precision (or recall) is the averaged precision (or recall) of all items:

$$\mathrm{P_{B3}} = \mathrm{Avg}_e[\mathrm{Avg}_{e' s.t. C(e)\cap C(e')\neq\emptyset}[\mathrm{P}(e,e')]]$$
$$\mathrm{R_{B3}} = \mathrm{Avg}_e[\mathrm{Avg}_{e' s.t. L(e)\cap L(e')\neq\emptyset}[\mathrm{R}(e,e')]]$$

6.3 Results

Figure 5 shows the F_β-score for various β values:

$$\mathrm{F}_\beta = \frac{(1+\beta^2)\cdot \mathrm{P_{B3}}\cdot \mathrm{R_{B3}}}{\beta^2\cdot \mathrm{P_{B3}} + \mathrm{R_{B3}}}$$

This graph gives us a trade-off between precision and recall ($\beta = 0$ is exact precision and $\beta \to \infty$ tends to exact recall).[9]

Each curve in Figure 5 represents a particular clustering method. We include three naive baselines:

ewnc: Each word in its own cluster

aw1c: All words in one cluster

Random: Each target word is assigned M random cluster id's in the range 1 to K, then translation sets are clustered with the CP algorithm.

The curves for K-Means clustering include one condition with monolingual features alone and two curves that include bilingual features as well.[10] The bilingual curves correspond to two different feature sets: the first includes only unigram features (t1), while the second includes both unigram and bigram features (t1t2).

Each point on an F_β curve in Figure 5 (including the baseline curves) represents a maximum over two

[7] An evaluation using purity and inverse purity (extended to overlapping clusters) has been omitted for space, but leads to the same conclusions as the evaluation using BCubed.

[8] The metric does include in this computation the relation of each item with itself.

[9] Note that we use the micro-averaged version of F-score where we first compute $\mathrm{P_{B3}}$ and $\mathrm{R_{B3}}$ for each source-word, then compute the average $\mathrm{P_{B3}}$ and $\mathrm{R_{B3}}$ over all source-words, and finally compute the F-score using these averaged $\mathrm{P_{B3}}$ and $\mathrm{R_{B3}}$.

[10] All bilingual K-Means experiments include monolingual features also. K-Means with *only* bilingual features does not produce accurate clusters.

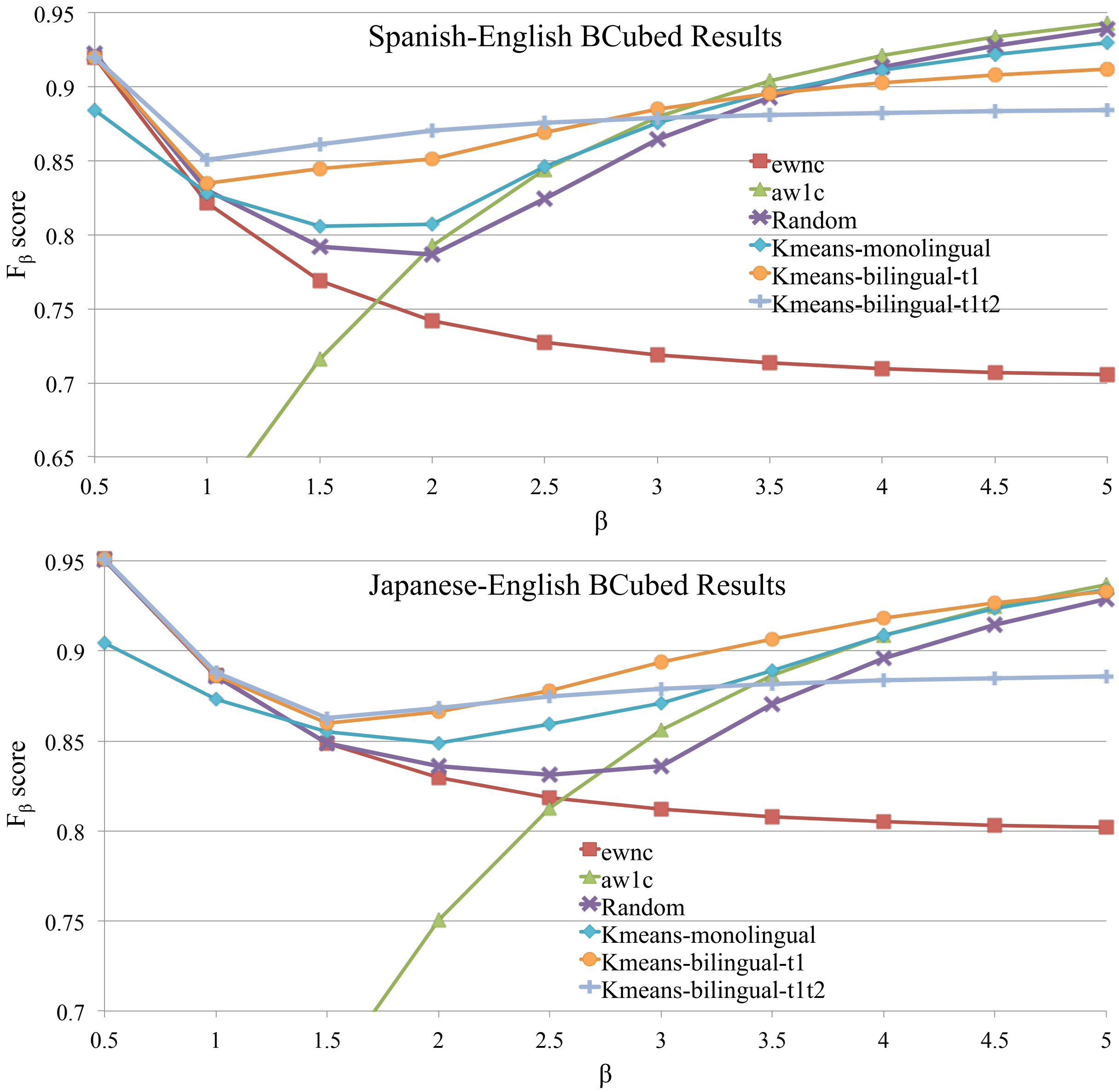

Figure 5: BCubed F_β plot for the Spanish-English dataset (top) and Japanese-English dataset (bottom).

Source word: *ayudar*		
Monolingual	[[*aid*], [*assist*, *help*]]	P=1.0, R=0.56
Bilingual	[[*aid*, *assist*, *help*]]	P=1.0, R=1.0

Source word: *concurso*		
Monolingual	[[*competition*, *contest*, *match*], [*concourse*], [*contest*, *meeting*]]	P=0.58, R=1.0
Bilingual	[[*competition*, *contest*], [*concourse*], [*match*], [*meeting*]]	P=1.0, R=1.0

Table 2: Examples showing improvements in clustering when we move from K-Means clustering with only monolingual features to clustering with additional bilingual features.

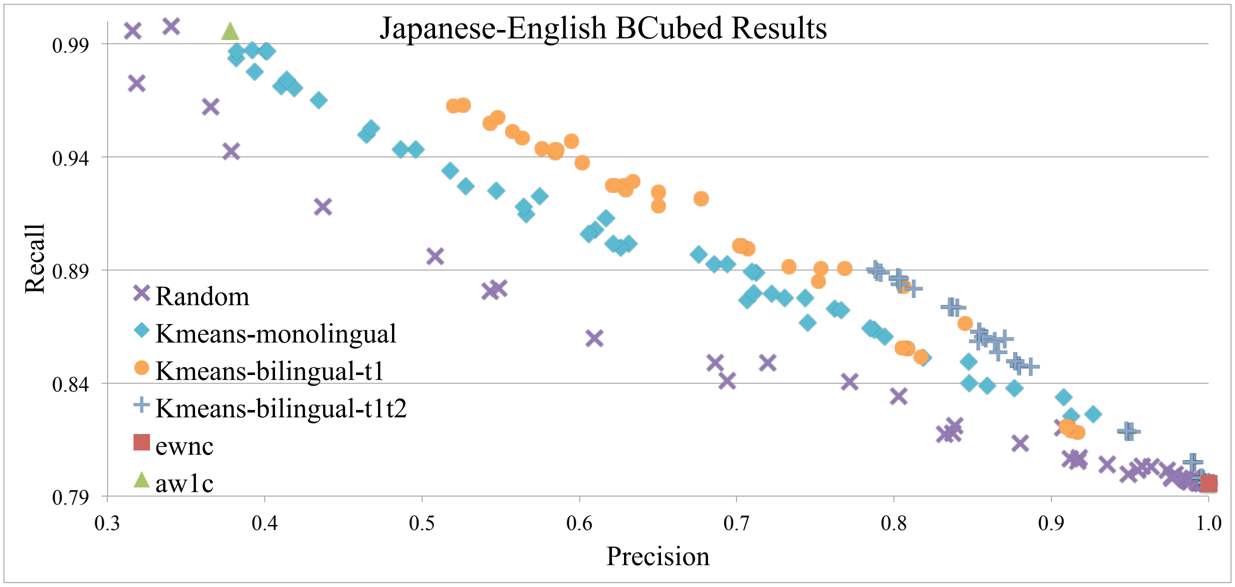

Figure 6: BCubed Precision-Recall scatter plot for the Japanese-English dataset. Each point represents a particular choice of cluster count K and clusters per word M.

parameters: K, the number of clusters created in the whole corpus and M, the number of clusters allowed per word (in M-best soft K-Means). As both the random baseline and proposed clustering methods can be tuned to favor precision or recall, we show the best result from each technique across this spectrum of F_β metrics. We vary β to highlight different potential objectives of translation sense clustering. An application that focuses on synonym discovery would favor recall, while an application portraying highly granular sense distinctions would favor precision.

Clustering accuracy improves over the baselines with monolingual features alone, and it improves further with the addition of bilingual features, for a wide range of β values. Our unsupervised approach with bilingual features achieves up to 6-8% absolute improvement over the random baseline, and is particularly effective for recall-weighted metrics.[11] As an example, in a S→E experiment with a K-Means setting of $K = 4096 : M = 3$, the overall $F_{1.5}$ score increases from 80.58% to 86.12% upon adding bilingual features. Table 2 shows two examples from that experiment for which bilingual features improve the output clusters.

The parameter values we use in our experiments are $K \in \{2^3, 2^4, \ldots, 2^{12}\}$ and $M \in \{1, 2, 3, 4, 5\}$. To provide additional detail, Figure 6 shows the BCubed precision and recall for each induced clustering, as the values of K and M vary, for Japanese-English.[12] Each point in this scatter plot represents a clustering methodology and a particular value for K and M. Soft K-Means with bilingual features provides the strongest performance across a broad range of cluster parameters.

6.4 Evaluation Details

Certain special cases needed to be addressed in order to complete this evaluation.

Target words not in WordNet: Words that did not have any synset in WordNet were each assigned to a singleton reference cluster.[13] The S→E dataset has only 2 out of 225 target types missing in WordNet and the J→E dataset has only 55 out of 1351 target

[11] It is not surprising that a naive baseline like random clustering can achieve a high precision: BCubed counts each word itself as correctly clustered, and so even trivial techniques that create many singleton clusters will have high precision. High recall (without very low precision) is harder to achieve, because it requires positing larger clusters, and it is for recall-focused objectives that our technique substantially outperforms the random baseline.

[12] Spanish-English precision-recall results are omitted due to space constraints, but depict similar trends.

[13] Note that certain words with WordNet synsets also end up in their own singleton cluster because all other words in their cluster are not in the translation set.

types missing.

Target words not clustered by K-Means: The K-Means algorithm applies various thresholds during different parts of the process. As a result, there are some target word types that are not assigned any cluster at the end of the algorithm. For example, in the J→E experiment with $K = 4096$ and with bilingual (t1 only) features, only 49 out of 1351 target-types are not assigned any cluster by K-Means. These unclustered words were each assigned to a singleton cluster in post-processing.

7 Identifying Usage Examples

We now briefly consider the task of automatically extracting usage examples for each predicted cluster. We identify these examples among the extracted phrase pairs of a parallel corpus.

Let P_s be the set of source phrases containing source word s, and let A_t be the set of source phrases that align to target phrases containing target word t. For a source word s and target sense cluster G, we identify source phrases that contain *s and* translate to all words in G. That is, we collect the set of phrases $P_s \cap \bigcap_{t \in G} A_t$. We use the same parallel corpus as we used to compute bilingual features.

For example, if we consider the cluster [*place*, *position*, *put*] for the Spanish word *colocar*, then we find Spanish phrases that contain *colocar* and also align to English phrases containing *place*, *position*, and *put* somewhere in the parallel corpus. Sample usage examples extracted by this approach appear in Figure 7. We have not performed a quantitative evaluation of these extracted examples, although qualitatively we have found that the technique surfaces useful phrases. We look forward to future research that further explores this important sub-task of automatically generating bilingual dictionaries.

debajo		
["below","beneath"]	→	debajo de la superficie (*below the surface*)
["below","under"]	→	debajo de la línea (*below the line*)
["underneath"]	→	debajo de la piel (*under the skin*)
休養		
["break"]	→	一生懸命 働い た から 休養 する の は 当然 です . (*I worked hard and I deserve a good break.*)
["recreation"]	→	従来 の 治療 や 休養 方法 (*Traditional healing and recreation activities*)
["rest"]	→	ベッド で 休養 する だけ で 治り ます . (*Bed rest is the only treatment required.*)
利用		
["application"]	→	コンピューター 利用 技術 (*Computer-aided technique*)
["use","utilization"]	→	土地 の 有効 利用 を 促進 する (*Promote effective use of land*)
引く		
["draw","pull"]	→	カーテン を 引く (*Draw the curtain*)
["subtract"]	→	A から B を 引く (*Subtract B from A*)
["tug"]	→	袖 を ぐいと 引く (*Tug at someone's sleeve*)

Figure 7: Usage examples for Spanish and Japanese words and their English sense clusters. Our approach extracts multiple examples per cluster, but we show only one. We also show the translation of the examples back into English produced by Google Translate.

8 Conclusion

We presented the task of translation sense clustering, a critical second step to follow translation extraction in a pipeline for generating well-structured bilingual dictionaries automatically. We introduced a method of projecting language-level clusters into clusters for specific translation sets using the CP algorithm. We used this technique both for constructing reference clusters, via WordNet synsets, and constructing predicted clusters from the output of a vocabulary-level clustering algorithm.

Our experiments demonstrated that the soft K-Means clustering algorithm, trained using distributional features from very large monolingual and bilingual corpora, recovered a substantial portion of the structure of reference clusters, as measured by the BCubed clustering metric. The addition of bilingual features improved clustering results over monolingual features alone; these features could prove useful for other clustering tasks as well. Finally, we annotated our clusters with usage examples.

In future work, we hope to combine our clustering method with a system for automatically generating translation sets. In doing so, we will develop a system that can automatically induce high-quality, human-readable bilingual dictionaries from large corpora using unsupervised learning methods.

Acknowledgments

We would like to thank Jakob Uszkoreit, Adam Pauls, and the anonymous reviewers for their helpful suggestions.

References

Enrique Amigo, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 12(4):461486.

Marianna Apidianaki. 2009. Data-driven semantic analysis for multilingual WSD and lexical selection in translation. In *Proceedings of EACL*.

A. Bagga and B. Baldwin. 1998. Entity-based cross-document coreferencing using the vector space model. In *Proceedings of COLING-ACL*.

J. Bakus, M. F. Hussin, and M. Kamel. 2002. A SOM-based document clustering using phrases. In *Proceedings of ICONIP*.

Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jenifer C. Lai, and Robert L. Mercer. 1992. Class-based *n*-gram models of natural language. *Computational Linguistics*, 18(4):467479.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*.

Mona Diab and Philip Resnik. 2002. An unsupervised method for word sense tagging using parallel corpora. In *Proceedings of ACL*.

B.E. Dom. 2001. An information-theoretic external cluster-validity measure. In *IBM Technical Report RJ-10219*.

M. Halkidi, Y. Batistakis, and M. Vazirgiannis. 2001. On clustering validation techniques. *Journal of Intelligent Information Systems*, 17(2-3):107–145.

Hiroyuki Kaji. 2003. Word sense acquisition from bilingual comparable corpora. In *Proceedings of NAACL*.

Reinherd Kneser and Hermann Ney. 1993. Improved clustering techniques for class-based statistical language modelling. In *Proceedings of the 3rd European Conference on Speech Communication and Technology*.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of NAACL*.

Dekang Lin and Xiaoyun Wu. 2009. Phrase clustering for discriminative learning. In *Proceedings of ACL*.

J. B. MacQueen. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*.

Sven Martin, Jorg Liermann, and Hermann Ney. 1998. Algorithms for bigram and trigram word clustering. *Speech Communication*, 24:19–37.

M. Meila. 2003. Comparing clusterings by the variation of information. In *Proceedings of COLT*.

George A. Miller. 1995. Wordnet: A lexical database for English. In *Communications of the ACM*.

Hwee Tou Ng, Bin Wang, and Yee Seng Chan. 2003. Exploiting parallel texts for word sense disambiguation: An empirical study. In *Proceedings of ACL*.

Andrew Rosenberg and Julia Hirschberg. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of EMNLP*.

Michael Steinbach, George Karypis, and Vipin Kumar. 2000. A comparison of document clustering techniques. In *Proceedings of KDD Workshop on Text Mining*.

Lin Sun and Anna Korhonen. 2011. Hierarchical verb clustering using graph factorization. In *Proceedings of EMNLP*.

Dan Tufis, Radu Ion, and Nancy Ide. 2004. Fine-grained word sense disambiguation based on parallel corpora, word alignment, word clustering and aligned wordnets. In *Proceedings of COLING*.

Jakob Uszkoreit and Thorsten Brants. 2008. Distributed word clustering for large scale class-based language modeling in machine translation. In *Proceedings of ACL*.

Jakob Uszkoreit, Jay Ponte, Ashok Popat, and Moshe Dubiner. 2010. Large scale parallel document mining for machine translation. In *Proceedings of COLING*.

C. Van Rijsbergen. 1974. Foundation of evaluation. *Journal of Documentation*, 30(4):365–373.

Mariano Velázquez de la Cadena, Edward Gray, and Juan L. Iribas. 1965. *New Revised Velázques Spanish and English Dictionary*. Follet Publishing Company.

Andreas Vlachos, Anna Korhonen, and Zoubin Ghahramani. 2009. Unsupervised and constrained Dirichlet process mixture models for verb clustering. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. HMM-based word alignment in statistical translation. In *Proceedings of the Conference on Computational linguistics*.

W. Xu, X. Liu, and Y. Gong. 2003. Document-clustering based on non-negative matrix factorization. In *Proceedings of SIGIR*.

Y. Zhao and G. Karypis. 2001. Criterion functions for document clustering: Experiments and analysis. In *Technical Report TR 01-40, Department of Computer Science, University of Minnesota, Minneapolis, MN*.