EXPLORING CONTEXT CLUSTERING FOR TERM TRANSLATION

Zhila A. (alisa_zh@mail.ru),
Gelbukh A. (gelbukh@gelbukh.com)

Center for Computing Research, Instituto Politécnico Nacional, Mexico City, Mexico

Many tasks in natural language processing, such as machine translation, word sense disambiguation, word translation disambiguation, require analysis of contextual information. In case of supervised approaches this analysis is performed by human experts, which is very costly. Unsupervised approaches offer fully automatic methods to fulfill these tasks. Yet these methods are not robust, their results are very parameter-dependent and difficult to interpret. Context clustering is an unsupervised technique for analysis of context similarities. In this work we explore dependencies of context clustering results from various clustering parameters. We also explore suitability of the context clustering for word translation disambiguation by evaluating the clustering results against known classes that are classes of translation candidates.

Key words: translation, translation candidates, clustering, unsupervised methods, parameter, word sense discrimination, context

1. Introduction

In natural language processing word sense disambiguation is the task of automatic assignment of a correct sense from a predetermined sense inventory to a polysemous word. It is tightly related to the task of machine translation, where a correct translation of a word or phrase must be chosen from a list of translation candidates. Recently, the task of selection of the best translation or several interchangeable (synonymous) translations for a given source word in context and a set of target candidates has become known as word translation disambiguation.

All these tasks require contextual information to resolve an ambiguity, albeit translational or semantic.

In natural language processing approaches that involve a manually tagged training corpus that is further used for training of a machine-learning algorithm are known as supervised methods. Methods that automatically learn from "raw" corpus are called unsupervised. There are also approaches that are based on manually crafted rules or use existing dictionaries or heuristics that are known according to [1] can be described as knowledge-based approaches.

In the past decade approaches to bootstrap machine translation with preliminary word sense disambiguation or word sense translation were explored in [23, 2–5].

These approaches are based on supervised WSD classifiers that require extensive training on a large manually tagged training corpus. They are resource-demanding and provide relatively little or no improvement at a high cost.

The task of word translation disambiguation was treated independently in [8, 12] either with a supervised classifier or with huge annotated monolingual corpora.

As it was noted in various reviews [1, 13], supervised methods have achieved substantial results but they require very costly training corpora, which are normally tagged by human experts. The training corpora have become a bottleneck of this approach and since its results anyway do not reach a human-made gold standard [7], ultimately the attention of researches has been driven to unsupervised methods.

There are two main directions in unsupervised methods: methods that use monolingual corpora and look for similarities in contexts or documents, as in context or document clustering, and methods that extract information from word aligned multilingual corpora also known as cross-lingual methods.

Context clustering is an unsupervised approach to detection of similarities in contexts [16, 20]. Its results highly depend on parameters used for clustering. This approach was applied to word sense discrimination [19], which is mere distinguishing between different senses.

Diab and Resnik [6] use cross-lingual approach for unsupervised word sense tagging. The authors use a word-aligned French-English parallel corpus with a tagged part in English to tag its French part with corresponding senses. This approach is aimed to facilitate sense-tagging of other languages given a broadly sense-tagged corpus in English. Consequently, although the suggested method is unsupervised, it requires substantially tagged data.

As follows from the above, the suitability of unsupervised approaches to word translation has not been explored. Our hypothesis is that unsupervised context clustering along with word aligned parallel texts can serve for obtaining context characteristics that would allow correct selection of a translation candidate for a word in a context in unsupervised manner. In this work we explore the suitability of context clustering for word translation disambiguation by comparing clustering results for various parameter combinations and evaluating them against known translation classes. In particular, we explore several parameter combinations with values that were found to be the best for the tasks of document and context clustering in [21, 24, 19, 11, 15]. For evaluation of clustering results we use translation equivalents that were obtained from word aligned parallel corpus.

The paper is organized as follows. In Section 2 we give a short overview of the parameters involved in context clustering. Section 3 describes experimental settings including context clustering software, dataset and the procedure for detection of dataset classes used for evaluation and interpretation. In Section 4 we demonstrate the obtained results and perform their analysis. Section 5 provides conclusion remarks and outlines future work in this direction.

2. Context Clustering

In the past decade the topic of unsupervised word sense discrimination, that is discrimination between different word usages *in context*, was actively investigated [1, 13]. The most known solution to this problem is clustering of contexts that contain a word in question, which is a particular application of document clustering [16, 20]. An extensive review of clustering as unsupervised classification of dataset elements into groups is provided in [9]. The clustering algorithms that are suitable for document clustering are described and analyzed in [21] and implemented first in CLUTO clustering toolkit [10], which receives an extension in SenseClusters clustering tool [18]. We adopted the latter as a tool for our experiments.

However, results of context clustering highly depend on a variety of parameters: clustering algorithms, criterion functions for cluster detection, context representations, context similarity measures, and cluster stopping criteria. Here we give a brief overview of clustering parameters and techniques.

2.1. Features

To perform a clustering one has to choose features that would represent each element of a dataset. In the field of document and context clustering each element, i. e. a document or a context, can be represented as a vector in a feature space. For example, a document can be represented as a vector of term frequencies:

$$\mathrm{dtf}=(\mathrm{tf}_1, \mathrm{tf}_2, \ldots, \mathrm{tf}_n),$$

where tf_i is the frequency of a particular term i in a document and n is the number of all terms from the entire document set.

Features are called *unigrams*, when only one-word terms are considered. Unigrams are considered to be quite a simple model that gives no information on possible word collocations. Nevertheless, they are proved to be useful for measuring first-order similarity (see Section 2.2) for short context with regular vocabulary, e. g. weather forecasts [15]. Moreover, they can be used for second-order similarity measurement (see Section 2.2) using Latent Semantic Analysis (LSA) representation. Pairs of two consecutive words are called *bigrams*. They are used in second-order similarity measurement, which proved to be more appropriate for short contexts that do not share many common words [2]. In this work we adopt the extension of bigram's definition that is introduced in [17] and is implemented in SenseClusters [18]. The extended definition states that bigrams are pairs of words that occur in a given order within some distance from each other. The distance is called *window*. For example, for a window of size five there could be at most three intervening words between the first and the second word that make up a bigram. The small window value represent narrow context in modeling, which for short context may result in lower similarity. However, larger windows might involve unrelated words that are never seen in collocations with smaller windows. Normally, the window is set between 2 and 10. In contrast to the

bigrams, unordered pairs of words within a given window are called *co-occurrences*. For contexts that contain a marked word as in the case of word sense discrimination, *target co-occurrences* are introduced. Target co-occurrences are co-occurrences that include the marked word. Such words as auxiliary verbs, articles, conjunctions, etc., that are common for any context and, therefore, do not bring in any characteristic information are known as stopwords and are not considered in features.

Moreover, words that occur fewer times than a threshold (*frequency-cut parameter*) *r* cannot serve as a solid basis for context grouping and, hence, must be excluded from the feature list as well. The typical value of the frequency-cut parameter *r* is between 3 and 5.

2.2. Order of context representation

The first-order representation represents a context as a vector only of those features that are directly present in the context. The second-order representation also considers features that co-occur with the initial context features in other context. For example, if we have context 1 *"computer mouse"* and context 2 *"wireless mouse"* with bigram features, they will not have any common features for the first-order representation. Yet, for the second-order representation, at the training phase of feature gathering, the contexts may serve for mutual extension of features, "wireless mouse" being a second-order feature for context 1 and "computer mouse" being a second-order feature for context 2. Pedersen [15] shows the second-order representation to be better for short contexts since they contain fewer words than a document. Both representations are implemented in SenseClusters toolkit.

2.3. Similarity measure

To evaluate similarity between contexts, a *similarity measure* must be introduced on the selected feature representation. If elements are represented as feature vectors, such similarity measures as distance or cosine can be used. Hence, contexts can be either represented in a *vector space*, where a vector corresponds to each document, or a *similarity matrix* can be constructed based on pairwise similarities between contexts.

2.4. Clustering criterion functions

The task of clustering is optimization of a clustering criterion function, which is a function from similarity measure. A review and comparison of criterion functions for partitional clustering is presented in [24]. The authors evaluate the performance of eight different criterion functions for the problem of document clustering. Internal criterion functions I_1, I_2, and I_3 are based on the intra-cluster (dis)similarity, while external criterion functions E_1 and E_2 consider inter-cluster distances or (dis)similarities.

Hybrid criterion functions H_1 and H_2 combine the properties of internal and external criterion functions. All these functions view a document as a feature vector. Graph based criterion functions G_1 and G_2 are based on graph representation of a document. The authors show that two of the compared criterion functions (I_2 and H_2) steadily provide good results with most of the clustering algorithms.

2.5. Clustering techniques

A variety of clustering techniques and algorithms exists to determine the sequence of steps for grouping and further regrouping of elements. This variety can be classified into three groups: hierarchical clustering, partitional clustering, and hybrid. A detailed description of these techniques and comparison of their application for document clustering can be found in Steinbach *et al*. [21]. As it follows from their research, the best clustering techniques for document clustering were bisecting k-means among partitional clustering techniques and refined agglomerative clustering with UPGMA (Unweighted Pair-Group Method with Arithmetic Mean) among hierarchical techniques. They also showed that bisecting k-means technique performed better than refined agglomerative clustering with UPGMA.

2.6. Cluster stopping criteria

However, the existing clustering techniques imply that a number of clusters is known in advance, which is not true in many cases, especially for context clustering and word sense discrimination. When we do not know a desired number of clusters, automatic cluster stopping criteria were suggested in [11]: *gap*, which is based on gap statistics applied to within-cluster dispersion, *pk1*, *pk2*, and *pk3* ("*pk*" stands for "Predicting the number of clusters K"), which consider significance of the change of a cluster similarity function between sequential number of clusters. In this work we use cluster stopping criteria to automatically detect the number of word senses of an investigated word based on the idea of unsupervised word sense discrimination that holds that similar senses are accompanied by similar contexts. We then compare the resulting number of clusters to the number of senses provided by dictionaries, by these means detecting the most appropriate cluster stopping criteria.

2.7. Clustering evaluation

There are many different quality measures for evaluation of clustering results. Ranking of clustering algorithm performance depends substantially on what measure is used [21].

There are two basic approaches to clustering evaluation: internal and external. Although the classification might coincide with cluster criterion functions, clustering evaluation measures take different perspective on clusters. Internal clustering quality

measures do not use any external knowledge about possible groupings of clustered elements. They are based on inter-cluster similarity or dissimilarity data, although unlike the external clustering criteria functions, evaluation functions use this information for clustering evaluation rather than optimization. External quality measures compare clustering results to an external set of known classes of the clustered elements. For example, for document clustering it can be a set of predetermined topics. The result of evaluation with an internal quality measure directly depends on the clustering function used for the clustering and such evaluation is difficult to interpret for a set of contexts from the point of view on their information content.

In our work we evaluate clustering results against the set of translation equivalents obtained from a parallel corpus (see Section 3). We chose the widespread external measures of entropy and purity implemented in SenseClusters toolkit. We adopt the definitions and formulas for clustering entropy and purity given in [21]. The entropy looks at uncertainty of a class distribution via clusters, while the purity evaluates how good a class corresponds to one cluster In brief, the lower is the entropy and the higher is the purity, the better.

3. Experimental Settings

This section describes the experiment that we performed. It is aimed at exploration of how well unsupervised context clustering forms clusters that would be appropriate for word translation disambiguation. For this experiment, first, we use SenseCluster toolkit to perform unsupervised context clustering on our dataset. Then, we evaluate the clustering results comparing them to the context classes formed by identity of corresponding translations from parallel texts. This experiment is a continuation and extension of works [19, 21, 24]. In contrast to our experiment, they all dealt only with monolingual material. Works [21, 24] detected some optimal parameters –clustering techniques and criterion functions respectively– for document clustering. They used collections of abstracts as their dataset. Hence, the sizes of the documents are comparable to a size of a short context that we use in the work. Therefore, their results are applicable for our experiment. Work [19] compares vector and similarity spaces for context clustering. For their experiment the authors used senseval-2 word sense disambiguation dataset, where each context is about 3–7 sentences per contexts, which perfectly correlate with the size of the contexts in our experiment.

We base our experiment on the results obtained in these work and explore whether these clustering parameters are appropriate for translation detection. The main assumption is that translations are correlated with a sense of a translated word. Unsupervised word sense discrimination holds that similar senses will be grouped into one cluster. Consequently, the contexts in a cluster would correspond to one translation equivalent or to several synonymous translations. We also expect that two or more clusters might correspond to the same translation, which is true for the case when the translation preserves the same homonymy as the translated word.

3.1. Dataset

We used sentence aligned English-Spanish Europarl parallel corpus from OPUS open corpus [22] to extract contexts for clustering and to detect translation equivalents.

For our purpose of exploring context clustering suitability for word translation disambiguation, an ambiguous word had to satisfy the following criteria:

- to have a number of instances in a chosen parallel corpus that would be sufficient for unsupervised clustering (we set it 1000, which is about 2 to 5 times more than senseval-2 datasets used in [19] and realistic enough to be extracted from a corpus);
- to have more than one candidate translation in the parallel part of a corpus.

The analysis of the above criteria was performed using OPUS word alignment database. We have chosen several words that satisfy these criteria: *facility*, *post*, *language*. Due to time constraints, we present results only for the word "FACILITY".

As a context we used an extract of 7 consecutive sentences from the corpus, a sentence with the chosen source word being the forth. We chose the seven sentence context size basing on the average lengths of senseval-2 contexts. If a sentence containing the target word were closer to the beginning or the end the size of a context remained the same with more sentences to the end or beginning correspondingly. At this step we extracted 1771 contexts for our dataset.

The dataset was converted to lower-case and tokenized.

For evaluation of clustering results we obtained a set of corresponding translation equivalents from the same parallel corpus. Initially we intended to perform word alignment automatically with alignment tool GIZA++ [14]. Yet we obtained excessively many word-to-NULL alignments for the chosen word. It might be due to a relatively small size of the dataset corpus, which additionally contained nearly 20 % (342) of wrong sentence alignments.

Therefore, we developed an alternative approach to detection of corresponding translations for a selected source word. For *ca.* 600 entries of our dataset, pruned alignments were available from OPUS word alignment database. The rest was detected manually by comparing source word contexts with their corresponding parallel contexts.

First, we deleted wrongly aligned contexts from our dataset.

There were also cases when the word "facility" did not have a direct translation equivalent. We tagged such cases as NOTAG since we wanted to detect whether unsupervised sense clustering would find something in common between contexts that are translated in this manner.

Further, we grouped low-frequency (from 1 to 6) translation equivalents with their synonyms considering their context usage.

In the end, we obtained a dataset of 1429 contexts and 21 translation classes including NOTAG. These translation classes serve for external evaluation of obtained clusters. The dataset along with a translation candidate key file and information on some intermediate steps can be found at *www.gelbukh.com/resources/word-translation-alignments*.

According to the monolingual dictionaries, which we consulted (Online Merriam-Webster, Oxford Concise Thesaurus, WordNet, and Larousse American Pocket), they distinguish between 4 and 5 senses for the word "facility" that can be described as:

- installation, building;
- service;
- equipment;
- possibility;
- readiness.

We took these numbers as guidance for the minimum number of clusters. Therefore, any combination of parameter values that gave fewer than 4 clusters was discarded from the comparison of parameter values.

3.2. Clustering parameters

In this work we perform context clustering with SenseClusters toolkit [18]. It is a complete and freely available context clustering system that provides support for feature selection from large corpora, several different context representation schemes, various clustering algorithms, and evaluation of the discovered clusters.

Parameters with fixed values We set values of several parameters to be unchangeable and regarded as "default" for our experiments:

- the order of feature representation is set to *-o2*, which stands for the second order;
- the context are represented as feature vectors in *vector space*;
- window is set to 5;
- frequency-cut parameter *r* is set to 3.

We chose the second-order context representation since it is shown to be better for short contexts [15], 7-sentece contexts being regarded as a short contexts.

The vector space is preferred over the similarity matrix representation based on the work of Purandare and Pedersen [19], where contexts of the similar size were used for the task of word sense discrimination. They analyzed 6 combinations of 4 clustering parameters: order of context representation, features, vector space/similarity matrix, and clustering method. Purandare and Pedersen show that the best results were achieved for combinations with vector space.

For the value of window parameter we took as a reference the work by Purandare [17], where this parameter was set to 5.

We set the value of the frequency-cut parameter *r* to 3 heuristically. We considered that in [17] it was set to 2 for datasets that were 2 to 5 times smaller than ours. Therefore, we slightly increased the frequency-cut parameter to avoid too much noise and yet we did not increase it significantly so that significant features would not be cut out.

Parameters with varied values In the experiment we varied several parameters: features for context representation, clustering methods, criterion functions and cluster stopping criteria. Since the total number of possible combinations is very high, we analyzed only among those parameter values that are proved to be the best for document and context clustering in [21, 24]. We also considered repeated bisections and refined repeated bisections methods since they are considered in works on context clustering [19, 20]. The parameters with their varied values are:

- features for context representation: unigrams, bigrams, co-occurrences, target co-occurrences;
- clustering methods: direct k-means, repeated bisection, refined repeated bisection, agglomerative;
- criterion functions: I_2 and H_2 for partitional methods and UPGMA for the agglomerative method;
- cluster stopping measures: gap, pk1, pk2, pk3.

The total number of experiments is 112.

4. Experimental results

The number of clusters that we obtained with various clustering parameter combinations varied from 1 to 6.

Cluster stopping measures. Table 1 shows the frequencies of each number of clusters for a cluster stopping measure.

Table 1. Distribution of resulting cluster number per cluster stopping measure.

cl. num.	1	2	3	4	5	6
gap	24	0	4	0	0	0
pk1	11	10	3	1	3	0
pk2	0	8	10	3	4	3
pk3	0	12	9	6	1	0

As it follows from Table 3, cluster stopping measures *gap* and *pk1* provide the lowest number of clusters. Gap statistic measure gives no results that would be higher than the threshold of 4 clusters. Pk1 measure gives acceptable results only in 4 cases, which is 3.5 % of all cases.

The fractions of experiments for each cluster number from the total number of experiments are shown in Table 2.

Table 2. Fraction of experiments that resulted in each cluster number.

cl. num.	1	2	3	4	5	6
fraction, %	31.2	26.8	23.2	9.0	7.1	2.7

Of the total number of experiments 50 % were for cluster numbers 2 and 3, and only 18.8 % (21 of a total of 112 combinations) passed the threshold of 4 clusters.

An assumption that the word "facility" may have only 2 to 3 "*real*" or well distinguishable senses does not seem to be probable. If we take a look at the list of generalized senses for "facility" in Section 3, they hardly can be grouped into a number of independent and non-intersecting senses less than four. And if we take into account that a lexical company of a word in context might vary even more than its semantic meaning, we would rather expect a larger number of clusters than a smaller one.

Therefore, we interpret the steadily low number of clusters for cluster stopping criteria gap and pk1 as an inherent quality of these criteria. *Pk2* and *pk3* measures give acceptable results in 36 % and 25 % of their usage cases respectively.

Context features, entropy and purity. The parameter values, the entropy, and the purity for the experiments, for which the number of clusters resulted to be at least 4 are presented in Table 3. We remind that the number of senses for "facility" given by monolingual dictionaries is 4 to 5.

It is to be noted that no experiments with bigram features, which are two consecutive words in a given window, and target co-occurrences, which are co-occurrences with the target word "facility", resulted in the number of clusters more than or equal to 4. Therefore, the results of all experiments with these features were discarded and are not shown in Table 3. The fact that clustering with bigram and target co-occurrence features gave very low numbers of resulting clusters might be explained by a hypothesis that conditions imposed on these features are hard to satisfy: bigrams require repeated consecutiveness of a word pair and target co-occurrences require co-occurrence with a target word within a certain window. Therefore, only contexts with very high frequency features are clustered together, and the remaining cluster used for contexts that could not be clustered with others. To check this explanation further experiments with lower frequency-cut value and wider window are needed.

Notations of Table 3 are as follows: *clmeth* stands for "clustering method", *crfun* stands for "criterion function", *clstop* stands for "cluster stopping measure", *cl #* is the resulting number of clusters for a given combination of parameters, *E* and *P* are entropy and purity respectively. In the table we used the following abbreviations: *agglo* for agglomerative clustering method, *direct* for direct k-means, *rb* for repeated bisection, *rbr* for refined repeated bisection. These notations and abbreviations are used in the rest of the paper. Other values that are expressed with alphanumeric sequences are explained in Section 2 and in Section 3.2.

Table 3. The best experiment results.

clmeth	crfun	clstop	cl #	E	P	clmeth	crfun	clstop	cl #	E	P
Co-occurrences						Unigrams					
agglo	upgma	pk2	6	80.6	25.5	agglo	upgma	pk2	6	84.1	24.2
direct	h2	pk1	4	80.4	25.6	direct	i2	pk2	6	**74.8**	26.9
direct	h2	pk3	4	80.4	25.6	rb	h2	pk1	5	75.2	**28.3**
direct	i2	pk2	5	80.2	25.5	rb	h2	pk2	4	76.2	27.6
direct	i2	pk3	4	80.4	25.6	rb	h2	pk3	4	76.2	27.6
rb	h2	pk1	5	80.7	25.0	rb	i2	pk2	5	75.6	27.8
rb	h2	pk2	4	81.0	25.0	rbr	h2	pk1	5	75.2	**28.3**
rb	h2	pk3	4	81.0	25.0	rbr	h2	pk2	4	76.2	27.6
rb	i2	pk3	5	80.7	25.0	rbr	h2	pk3	4	76.2	27.6
rbr	h2	pk3	4	80.4	25.6	rbr	i2	pk2	5	75.3	**28.3**
rbr	i2	pk2	5	80.2	25.5						

As it can be observed from Table 3, several parameter combinations with unigram and co-occurrence features passed the threshold. It can be observed from Table 5 that for partitional clustering techniques –direct k-means, repeated bisections and refined repeated bisections– variation of entropy and purity has some dependency on the number of clusters. For fixed number of clusters and context feature pairs of entropy and purity can be grouped into as few as one or two groups of equal values. For example, if we set a context feature to be co-occurrence and a number for clusters to be 4, in 4 of 6 cases (entropy; purity) = (80.4; 25.6) and in the rest of the cases (entropy; purity) = (81.0; 25.0). To detect the actual dependence further experiments are needed.

The best entropy and purity values correspond to the parameter combinations with unigram features. In general, the entropy for unigrams is about 5% better than the entropy for co-occurrences and the purity is 12% better for unigrams than for co-occurrences. Yet comparison of these entropy and purity values to those obtained in [21, 24] is hindered by the dependence of entropy and purity on the number of classes.

Our consideration is that the entropy and purity measures as they are described in Section 2.7 might be inappropriate for cluster evaluation in our task. These measures were intent to evaluate word sense discrimination results, when it is assumed that each cluster corresponds to a sense and it is expected (or manually set) that the number of clusters would be more or less the same as the number of senses. On the contrary, in our case it is completely acceptable if more than one class are clustered together, which corresponds to the case of synonymous translations, or if elements of one class are distributed between several clusters, which is the case of preserved homonymy.

Number of clusters. To check how cluster number will influence the entropy and purity, we performed an experiment with the number of clusters manually set to 21, which is the number of our translation classes. In this experiment we used a clustering parameter combination that gave the highest purity. The results are shown in Table 4.

Table 4. Entropy and purity for a predetermined number of clusters.

clmeth	crfun	clstop	cl #	E	P
Unigrams with fixed number of clusters					
rb	h2	n/a	21	67.2	32.7

As it can be seen, the more than fourfold increase of the cluster number from 5 to 21 improves the values of entropy and purity only 10.6 % and 15.5 % respectively.

Illustration of clustering results. For illustration of clustering results we chose two of the best experiment cases: one for the co-occurrence feature and another for the unigram feature. We assume that this small sample from the set of 112 experiments would be enough demonstrative to provide general impression of the experiment and will not consume much space.

Table 5 shows the distribution of translation classes through resulting context clusters for the chosen parameter combinations.

Table 5. Illustration of the class-cluster distribution for two of the best experiment cases.

clD	cl. size	disp	equi	Faci	instal	*NO*	serv	sist	posib	meca	instit	capac	centro	ayud	medio	crédi	planta	fond	medida	infra	plan	central
	Co-occurrences, Agglomerative Clustering, UPGMA, Pk2, Entropy = 0.806, P= 0.255																					
0	989	42	22	121	196	94	96	19	95	82	16	25	18	19	55	18	16	20	8	14	4	9
1	3	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
2	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	350	3	6	24	142	40	40	3	14	6	10	4	9	0	20	1	6	1	5	2	0	14
4	11	0	0	5	1	1	0	0	2	0	0	0	1	0	0	0	0	0	0	1	0	0
5	75	18	1	1	4	3	5	19	4	8	2	0	1	2	1	1	0	5	0	0	0	0
	Unigrams, Repeated Bisections, H2, Pk1, E= 0.758, P=0.276																					
0	213	27	1	24	8	17	16	6	24	26	1	9	3	7	12	9	1	15	1	3	2	1
1	156	1	0	1	112	10	3	0	1	1	0	0	1	0	5	0	8	0	2	0	0	11
2	282	0	13	42	83	26	35	3	22	4	6	4	8	3	13	1	3	0	2	9	0	5
3	307	32	1	29	24	25	11	25	32	57	4	9	6	7	18	7	3	10	4	0	2	1
4	471	3	14	56	116	61	77	7	36	8	17	7	11	4	28	4	7	1	4	5	0	5

To make it easier to interpret, we give an illustration of several consecutive contexts (shortened to one sentence with the target word to save some space) that were clustered together in cluster 0 in the experiment with the second parameter combination from Table 5. A corresponding Spanish translation is located between quotes in a translation id tag and a short definition in English is given in brackets after the Spanish translation:

<translation id="dispositivo" (mechanism; device)>

Finally, we need some coordination of national maritime authorities in order to achieve some sort of European facility comparable to the coastguards who supervise the coasts of the United States.

<translation id="capacidad" (capacity; here: creditworthiness)>

We shall also vote against the abolition of the facility for the Member States to increase to 35%...

<translation id="servicio" (service)>

It would be a serious matter if every non-European company were to learn to use Europe as a bus where you do not have to pay for the ticket, you do not have to pay for cooperation, you benefit from using the facility and you leave without being accountable to anyone.

Here we have an example of one of 27 contexts tagged with the translation "dispositivo", one of 9 "capacidad" contexts, and one of 16 "servicio" contexts that all were clustered together in cluster 0 along with other contexts corresponding to all translation variants.

In the beginning of Section 3 we explained the assumption that unsupervised context clustering would be suitable for word translation selection if a cluster corresponded to one or more entire translation classes, which is the case of synonymy between translations, or if a translation class was distributed between *some* clusters, which is the case of preserved cross-lingual homonymy. For these cases a clustering solution would tend to obtain "neat" groupings of classes per clusters and intuitively we can predict that a class-cluster distribution table would have more zeros than non-zeros. However, we see (especially in the unigram case) that nearly all cells have non-zero values. It means that a context corresponding to any translation equivalent can be found in any cluster. This violates our initial assumption about context clustering suitability for word translation selection.

5. Conclusions and future work

In this work we perform comparison of various clustering parameter combinations and explored suitability of context clustering application to unsupervised word translation.

The number of clusters more than the threshold of 4 occurred only for 18.8% of the experiments. Numbers of 2 and 3 were detected in 50% of cases. Yet these results cannot be interpreted from the semantic point of view, therefore, they were discarded as it was initially intended. However, formal analysis of semantic similarity of senses through an ontology or semantic hierarchy can give new perspective on these numbers.

We detected that cluster stopping measures gap and pk1 provide very low numbers of clusters that cannot be interpret from the semantic point of view. The numbers of clusters that correspond to the semantic assumption of the number of word senses can be achieved in most cases with pk2 and pk3 cluster stopping measure. Also pk1 cluster stopping measure should not be completely discarded since it provided 19% of all acceptable results.

We were not able to detect acceptable results for bigram and target co-occurrence features. It might be explained by inappropriate window size and data sparseness that in our experiments was not handled through singular value decomposition. Hence, further experiments with singular value decomposition and varying window size are necessary.

The evaluation of results through entropy and purity gives us the numbers that are not easily interpreted in the task of word translation when the number of classes is much higher than the number of clusters. Hence, we will work on development of different quality measure that would be more adequate for our goals.

References

1. *Agirre E., Edmonds P.* (eds.) (2006), Word Sense Disambiguation. Algorithms and Applications, Springer.
2. *Carpuat M., Wu D.,* Word sense disambiguation vs. statistical machine translation. Proc. of the annual meeting of the ACL, 2005, pp. 387–394.
3. *Carpuat M., Wu D.*, Evaluating the word sense disambiguation performance of statistical machine translation. Proc. of the Second International Joint Conference on Natural Language Processing (IJCNLP), 2005, pp.122–127.
4. *Carpuat M., Wu D.*, Improving statistical machine translation using word sense disambiguation, Proc. of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007), 2007, pp. 61–72.
5. *Chan Y. S., Ng H. T.*, Word sense disambiguation improves statistical machine translation, Proc. of the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, 2007, pp. 33–40.
6. *Diab M., Resnik P.*, An unsupervised method for word sense tagging using parallel corpora, Proc. of the 40th Annual Meeting on Association for Computational Linguistics, 2002, pp. 255–262.
7. *Gale W., Church K. W., David Yarowsky D.* Estimating upper and lower bounds on the performance of word-sense disambiguation programs. Proc. of the 30th Annual Meeting of the Association for Computational Linguistics, Newark, Delaware, 1992.
8. *Holmqvist M.*, Memory-based learning of word translation, Proc. of the 16th Nordic Conference of Computational Linguistics NODALIDA-2007, Tartu, Estonia, 2007, pp. 231–234.
9. *Jain A. K., Murty M. N., Patrick J. Flynn P. J.* (1999), Data Clustering: A Review. ACM Computing Syrveys, vol. 21, pp. 264–323.
10. *Karypis, G.* (2003), CLUTO — A Clustreing Toolkit, University of Minnesota, Department of Computer Science Technical Report 02–017.
11. *Kulkarni A., Pedersen, T.* (2006), Unsupervised Context Discrimination and Automatic Cluster Stopping, MS Thesis, University of Minnesota Supercomputing Institute Research Report UMSI 2006/90.

12. *Marsi E., Lynum A., Bungum L., Gambäck B.*, Word Translation Disambiguation without Parallel Texts. Proc. International Workshop on Using Linguistic Information for Hybrid Machine Translation, Barcelona, Spain, 2011.
13. *Navigli R.* (2009), Word sense disambiguation: A survey, ACM Computing Surveys, vol. 41(2), pp. 1–69.
14. *Och F. J., Ney H.* (2003), A Systematic Comparison of Various Statistical Alignment Models, Computational Linguistics, vol. 29(1), pp. 19–51.
15. *Pedersen T.* (2008), Computational Approaches to Measuring the Similarity of Short Contexts: A Review of Applications and Methods, University of Minnesota Supercomputing Institute Research Report UMSI 2010/118.
16. *Pedersen T., Bruce R.*, Distinguishing word senses in untagged text, Proc. of the Second Conference on Empirical Methods in Natural Language Processing, Providence, RI, 1997, pp. 197–207.
17. *Purandare A.* (2004), Unsupervised Word Sense Discrimination By Clustering Similar Contexts. MS Thesis. University of Minnesota.
18. *Purandare A., Pedersen T.*, SenseClusters — Finding Clusters that Represent Word Senses. Proc. of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-04), 2004, pp. 26–29.
19. *Purandare A., Pedersen T.*, Word Sense Discrimination by Clustering Contexts in Vector and Similarity Spaces, HLT-NAACL 2004 Workshop: Eighth Conference on Computational Natural Language Learning (CoNLL-2004), 2004, pp. 41–48.
20. *Schütze, H.* (1998), Automatic Word Sense Discrimination. Journal of Computational Linguistics, vol. 24(1), pp. 97–123.
21. *Steinbach M., Karypis G., Kumar V.* (2000), A comparison of document clustering techniques, University of Minnesota, Technical Report 00–034.
22. *Tiedemann J.* (2009), News from OPUS — A Collection of Multilingual Parallel Corpora with Tools and Interfaces, Recent Advances in Natural Language Processing, vol. V, pp. 237–248.
23. *Vickrey D., Biewald L., Teyssier M., Koller D.*, Word-sense disambiguation for machine translation. Proc. of the conference on Human Language Technology and Empirical Methods in Natural Language Processing 2005, 2005, pp. 771–778.
24. *Zhao Y., Karypis G.* (2001), Criterion Functions for Document Clustering: Experiments and Analysis, University of Minnesota, Department of Computer Science Technical Report 01–040.