Lang Resources & Evaluation (2013) 47:579–605
DOI 10.1007/s10579-012-9205-0

ORIGINAL PAPER

Evaluating Word Sense Induction and Disambiguation Methods

Ioannis P. Klapaftis · Suresh Manandhar

Published online: 2 March 2013

Abstract Word Sense Induction (WSI) is the task of identifying the different uses (senses) of a target word in a given text in an unsupervised manner, i.e. without relying on any external resources such as dictionaries or sense-tagged data. This paper presents a thorough description of the SemEval-2010 WSI task and a new evaluation setting for sense induction methods. Our contributions are two-fold: firstly, we provide a detailed analysis of the Semeval-2010 WSI task evaluation results and identify the shortcomings of current evaluation measures. Secondly, we present a new evaluation setting by assessing participating systems' performance according to the skewness of target words' distribution of senses showing that there are methods able to perform well above the Most Frequent Sense (*MFS*) baseline in highly skewed distributions.

Keywords Word Sense Induction · Word Sense Disambiguation · Lexical Semantics

1 Introduction

Word Sense Induction seeks to automatically identify the senses or uses of a given target word directly from a corpus (Brody and Lapata 2009). It is also known as unsupervised Word Sense Disambiguation, since WSI methods automatically create a sense inventory and disambiguate the ambiguous instances of a given word without relying on any external resources such as dictionaries or sense-tagged data.

I. P. Klapaftis (✉)
Microsoft Corporation, Redmond, WA, USA
e-mail: ioannisk@microsoft.com

S. Manandhar
Department of Computer Science, University of York, York, UK
e-mail: suresh@cs.york.ac.uk

Table 1 WSI example with four contexts of the target word *mouse*

ID	Induced sense	Contexts
A	S_1	The *mouse* is also used a lot in scientific research though it is not an easy animal to examine
B	S_2	Some *mouse* designs work like a joystick and may help. You can also use a touchpad ...
C	S_1	*Mice* are great animals for several reasons. They are small, inexpensive,...
D	S_2	I've been trying to install a new *mouse* on my touchpad but I have not succeeded yet...

Table 1 shows four contexts for the target word *mouse*. As can be observed, *mouse* appears with two senses, i.e. as a device in contexts *B*, *D* and as an animal in contexts *A*, *C*. The aim of a potential WSI system is to group the contexts of that target word into two clusters, so that each cluster contains only the target word contexts that refer to the same sense (second column of Table 1).

The main motivation for developing sense induction methods comes from the need to overcome the limitations of manually-constructed lexical databases such as WordNet (Fellbaum 1998) or OntoNotes (Hovy et al. 2006). In these databases, word senses are usually represented as a fixed-list of definitions. There are several disadvantages associated with the fixed-list of senses paradigm.

Firstly, machine-readable dictionaries suffer from the lack of explicit semantic, topical or contextual relations between concepts (Agirre et al. 2001). For instance, WordNet does not relate *cigarette* with *cancer*, although one would expect to find these two words co-occurring frequently.

Secondly, lexical databases often contain general definitions and miss many domain specific senses (Lin and Pantel 2002). For example, the definition of the first OntoNotes sense for the verb *connect*, i.e. *physically link or join two or more people, things, or parts*, is general enough, to include any object that can be connected to any other object. Such general definitions would possibly have a negative impact on Information Retrieval (IR) and Machine Translation (MT) applications that exploit word senses to semantically enhance their corresponding tasks. Similarly, the word *snood* is monosemous in WordNet and defined to be *an ornamental net in the shape of a bag that confines a woman's hair*. A simple web search for that word reveals that *snood* might also refer to a popular puzzle video game.[1]

Another important limitation of machine-readable dictionaries is that they often do not reflect the exact content of the context, in which the target word appears (Véronis 2004). For instance, the word *drug* in FrameNet (Baker et al. 1998) is defined to be *a chemical that affects the nervous system causing changes in perception*. However, depending on the context in which that word appears, i.e. a medical one, it is possibly beneficial to distinguish between the illegal narcotic and the medicine uses of *drug*.

[1] http://en.wikipedia.org/wiki/Snood_(video_game) [Access:09/12/2011].

A large part of work has been devoted on improving and enriching current sense inventories to deal with the aforementioned limitation. For instance, *Topic Signatures* (Agirre et al. 2001; Agirre and De Lacalle 2004) have been used to associate each sense entry with a list of topically related words. These words were derived by the web following a two-stage process. In the first stage, a query containing the monosemous relatives of a WordNet synset was sent to a commercial search engine and the retrieved web documents were downloaded. In the second stage, the downloaded documents were processed, words were extracted and weighted using χ^2 or *TF.IDF*.

Topic Signatures were further exploited in (Agirre and De Lacalle 2003) to cluster WordNet senses and create a more coarse-grained sense inventory, as well as in (Alfonseca and Manandhar 2002) for the purpose of extending WordNet with new unknown concepts. In the same vein, Kilgarriff et al. (2010) use distributional similarity to automatically create from a corpus a complete account of a word's grammatical and collocational properties having as a point of comparison the Oxford Collocations Dictionary.[2]

While all of the above approaches have shown to improve some of the limitations of hand-constructed lexicons, they are still based on the fixed-list of senses paradigm, in effect being unable to automatically create a sense inventory or model the usage of a particular word with respect to a given domain or application. WSI aims to overcome these limitations. In this paper, we present a thorough description of the SemEval-2010 WSI task (Manandhar et al. 2010), as well as an extension of the evaluation scheme used in the task. The description includes: (1) the methodology followed for constructing the publicly available datasets, (2) the participating teams, (3) the evaluation framework and (4) a comparative analysis of systems results. In the last part of our work, we extend the SemEval-2010 WSI evaluation setting by assessing sense induction methods both in an unsupervised and supervised manner according to the skewness of the distribution of senses for each target word.

The rest of the paper is structured as follows: Section 2 provides an overview of the current-state-of-the-art in sense induction and discusses the evaluation setting used in SemEval-2007 WSI task (Agirre and Soroa 2007a). Section 3 describes the SemEval-2010 WSI task and summarises the methods of participating systems. Section 4 describes the evaluation framework of the task and provides an analysis of participating systems' results. Section 5 evaluates WSI methods on a new evaluating scheme and finally, the last section summarises our work providing an outlook on future work.

2 Background

2.1 Overview of sense induction methods

Word Sense Induction methods can be broadly divided into three categories, i.e. vector-based, graph-based and Bayesian methods. Most of the work in WSI is based

[2] http://elt.oup.com/teachers/ocd/ [Access:09/12/2011].

Table 2 Example word vectors

ContextID/Dimension	Research	Animal	Design	Joystic	Touchpad
A	1	1	0	0	0
B	0	0	1	1	1
C	0	1	0	0	0
D	0	0	0	0	1

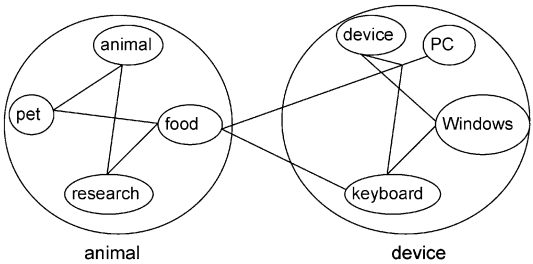

Fig. 1 Graph example for the target word *mouse*

on the Vector Space Model (Salton and Buckley 1988). Typically, each instance (context) of a target word is represented as a vector of features (e.g. first or second-order word co-occurrences).

Table 2 shows an example of four context vectors for the contexts in Table 1. In Table 2, nouns excluding the target word were selected as dimensions of the vector space. We have applied binary weighting, i.e. each component of a context vector is weighted either with 1 when the word (feature) appears in a context and with 0 otherwise.

The resulting vectors are then clustered to produce the induced senses, where each sense might be a cluster of target word contexts (Schütze 1998; Purandare and Pedersen 2004; Pedersen 2007; Niu et al. 2007; Pinto et al. 2007), or a cluster of contextually related words (Lin and Pantel 2002).

Graph-based methods (Dorow and Widdows 2003; Véronis 2004; Agirre et al. 2006b) represent each word *w* co-occurring with the target word *tw* as a vertex. Two vertices are connected via an edge if they co-occur in one or more contexts of *tw*. Figure 1 shows an example of such a graph for the target word *mouse*.

Once the co-occurrence graph of *tw* has been constructed, different graph clustering algorithms are applied to induce the senses. Each cluster (induced sense) consists of a set of words that are semantically related to the particular sense. In the example of Fig. 1, a graph clustering method should produce two clusters corresponding to the two different senses of *mouse*.

Bayesian methods were recently applied to the task of sense induction. For instance, Brody and Lapata (2009) presented a sense induction method that is related to Latent Dirichlet allocation (LDA) (Blei et al. 2003). In their work, they model the target word instances as samples from a multinomial distribution over

senses which are in turn represented as distributions over words (Brody and Lapata 2009). The topics learned from their model correspond to the different senses of a given target word.

Klapaftis and Manandhar (2010) developed an unsupervised method for inferring the hierarchical grouping of the senses of a polysemous word. Their method constructs a graph, in which vertices are the contexts of a polysemous word and edges represent the similarity between contexts. The method of Hierarchical Random Graphs (Clauset et al. 2008) is then applied, in order to infer the hierarchical structure (binary tree) of the constructed graph.

2.2 Overview of SemEval-2007 WSI task

The first effort to evaluate WSI methods under a common framework (evaluation schemes and dataset) was undertaken in the SemEval-2007 sense induction task (Agirre and Soroa 2007a) that evaluated WSI methods on 35 target nouns and 65 target verbs. For each target word (noun or verb), participating teams were required to identify the senses of that word (e.g. as clusters of target word instances, co-occurring words, etc.), and secondly tag the target word instances using the automatically induced clusters. The output of a sense induction method was a list of target word instances, each one associated with an induced cluster.

For each target word the input corpus provided to participating teams consisted of texts from the Wall Street Journal. Evaluation was performed on a version of the input corpus tagged with OntoNotes (Hovy et al. 2006) senses. The evaluation scheme consisted of two settings, i.e. *unsupervised evaluation* and *supervised evaluation* described in the next section.

2.2.1 SemEval-2007 unsupervised evaluation

The aim of the unsupervised evaluation was to assess WSI methods in a similar fashion to Information Retrieval exercises using F-Score, i.e. the harmonic of precision and recall. The precision of a class G_i with respect to a cluster C_j is defined as the number of their common instances divided by the total cluster size, i.e. $P(G_i, C_j) = \frac{a_{ij}}{|C_j|}$. Similarly, the recall of a class G_i with respect to a cluster C_j is defined as the number of their common instances divided by the total sense size, i.e. $R(G_i, C_j) = \frac{a_{ij}}{|G_i|}$. Recall and precision can then be combined to produce the F-Score of a class with respect to a cluster ($F(G_i, C_j)$).

Given that a class can be associated with more than one clusters, the final F-Score ($F(G_i)$) assigned to class G_i is the maximum $F(G_i, C_j)$ value attained at any cluster C_j. Finally, the F-Score of the entire clustering solution is defined as the weighted average of the F-Scores of each GS sense (Eq. 1). In Eq. 1, m refers to the number of GS senses, while N is the total number of target word instances. If the clustering is identical to the original classes in the dataset, F-Score will be equal to one. In the example of Table 3, F-Score is equal to 0.714.

Table 3 Induced clusters and gold standard senses matrix

	G_1	G_2	G_3
C_1	500	100	100
C_2	100	500	100
C_3	100	100	500

$$F\text{-}Score(K,G) = \sum_{i=1}^{m} \frac{|G_m|}{N} F(G_m) \quad (1)$$

F-Score attempts to assess the quality of a clustering solution by considering two different angles, i.e. homogeneity and completeness (Rosenberg and Hirschberg 2007). Homogeneity refers to the degree that each cluster consists of data points which primarily belong to a single gold standard class. On the other hand, completeness refers to the degree that each gold standard class consists of data points which have primarily been assigned to a single cluster. A perfect homogeneity would result in a precision equal to 1, while a perfect completeness would result in a recall equal to 1.

Rosenberg and Hirschberg (2007) have shown that F-Score suffers from the matching problem which manifests itself either by not evaluating the entire membership of a cluster, or by not evaluating every cluster. The former situation is present, due to the fact that F-Score does not consider the make-up of the clusters beyond the majority class (Rosenberg and Hirschberg 2007). For example in Table 4, the F-Score of the clustering solution is 0.714 and equal to the F-Score of the clustering solution shown in Table 3, despite the fact that these are two different clustering solutions.

Specifically, the clustering in Table 4 has a better homogeneity than the clustering in Table 3, since each cluster contains fewer classes. Additionally, the second clustering has a better completeness since each gold standard class contains fewer clusters. The inability of F-Score to capture the difference in homogeneity and completeness between different clusterings has also been shown and confirmed in (Amigó et al. 2009). An additional instance of the matching problem of F-Score manifests itself, when it fails to evaluate the quality of smaller clusters, since these might not get mapped to a gold standard class. This might happen, when the clustering solution generates some clusters that only group a small number of target word instances.

In the SemEval-2007 WSI task (Agirre and Soroa 2007a), there were no systems able to perform better than the one-cluster-per-word (*1Cl1W*) baseline which groups all of the instances of a target word into one cluster. Additionally, systems that were able to perform close to that baseline did not perform well in the supervised

Table 4 Induced clusters and gold standard senses matrix

	G_1	G_2	G_3
C_1	500	0	200
C_2	200	500	0
C_3	0	200	500

evaluation scheme, since they were generating a very small number of clusters, in effect being biased towards the *1Cl1W* baseline.

2.2.2 SemEval-2007 supervised evaluation

In the supervised evaluation, the target word corpus is split into a testing and a mapping part. The mapping part is used to apply a soft probabilistic mapping of the automatically induced clusters to gold standard senses. In the next step, the testing corpus is used to evaluate WSI methods in a WSD setting.

For example, let us assume that the matrix shown in Table 3 has been produced by using the mapping part of the corpus. Table 3 shows that C_1 is more likely to be associated with G_1, C_2 is more likely to be associated with G_2 and C_3 is more likely to be associated with G_3. This information from the mapping part is utilized so as to create a matrix M, in which each entry depicts the conditional probability $P(G_i|C_j)$ (Table 5).

Given a new instance I of the target word from the testing corpus, a row cluster vector IC is created, in which each entry k corresponds to the score assigned to C_k to be the winning cluster of instance I. The product of IC and M provides a row sense vector IG, in which the highest scoring entry a denotes that G_a is the winning sense. For example, if we produce the row cluster vector $[C_1 = 0.8, C_2 = 0.1, C_3 = 0.1]$ and multiply it with the matrix of Table 5, then we get the row sense vector $[G_1 = 0.6, G_2 = 0.2, G_3 = 0.2]$ in which G_1 is the winning sense.

The supervised evaluation seems to favor WSI methods producing a higher number of clusters than the number of gold standard senses. This is due to the fact that clusters are mapped into a weighted vector of senses, and therefore inducing a number of clusters similar to the number of senses is not a requirement for good results (Agirre and Soroa 2007a). Despite that, a large number of clusters might also lead to an unreliable mapping of clusters to gold standard senses.

In the SemEval-2007 WSI task (Agirre and Soroa 2007a), an additional supervised evaluation of WSI methods using a different mapping/testing split than the official one resulted in a significantly different ranking of systems, in which all of the systems outperformed the *MFS* baseline. This result indicated that the supervised evaluation might not provide a reliable estimation of WSD performance, particularly in the case where the mapping relies on a single dataset split.

3 SemEval-2010 Task description

Figure 2 provides an overview of the Semeval-2010 task (Manandhar et al. 2010). As shown, the task consisted of three separate phases. In the first phase, *training*

Table 5 Mapping induced clusters to gold standard senses

	G_1	G_2	G_3
C_1	0.714	0.142	0.142
C_2	0.142	0.714	0.142
C_3	0.142	0.142	0.714

phase, participating systems were provided with a training dataset that consisted of a set of target word (noun/verb) instances (sentences/paragraphs). Participants were then asked to use this training dataset to induce the senses of the target word. No other resources were allowed with the exception of NLP components for morphology and syntax.

In the second phase, *testing phase*, participating systems were provided with a testing dataset that consisted of a set of target word (noun/verb) instances (sentences/paragraphs). Participants were then asked to tag (disambiguate) each testing instance with the senses induced during the *training phase*.

In the third and final phase, the tagged test instances were received by the organizers in order to evaluate the answers of the systems in a supervised and an unsupervised framework. Table 6 shows the total number of target word instances in the training and testing set, as well as the average number of senses in the gold standard.

The main difference of the SemEval-2010 against the SemEval-2007 sense induction task is that the training and testing data are treated separately, i.e the testing data are only used for sense tagging, while the training data are only used for sense induction. Treating the testing data as new unseen instances ensures a realistic evaluation that allows us to evaluate the clustering models of each participating system. Note however, that one of the participating teams (*Duluth-WSI*) used both the training dataset and the untagged version of the testing dataset to induce the senses. Using the untagged version of the testing dataset is likely to lead to an improved performance as opposed to using only the training data. This has been observed in (Agirre et al. 2006a) who extensively evaluated and optimised the parameters of *HyperLex*, a graph-based WSI method due to Véronis (2004).

3.1 Training dataset

The target word dataset consisted of 100 words, i.e. 50 nouns and 50 verbs. The training dataset for each target noun or verb was created by following a web-based semi-automatic method, similar to the method for the construction of *Topic Signatures* (Agirre et al. 2001). Specifically, for each WordNet (Fellbaum 1998) sense of a target word, we created a query of the following form:

$$< Target\ Word > \textbf{\textit{AND}} < Relative\ Set >$$

The *<Target Word>* consisted of the target word stem. The *<Relative Set>* consisted of a disjunctive set of word lemmas that were related to the target word sense for which the query was created. The relations considered were WordNet's hypernyms, hyponyms, synonyms, meronyms and holonyms. Each query was manually checked by one of the organizers to remove ambiguous words. The example in Table 7 shows the query created for the first[3] and second[4] WordNet sense of the target noun *failure*.

[3] An act that fails.

[4] An event that does not accomplish its intended purpose.

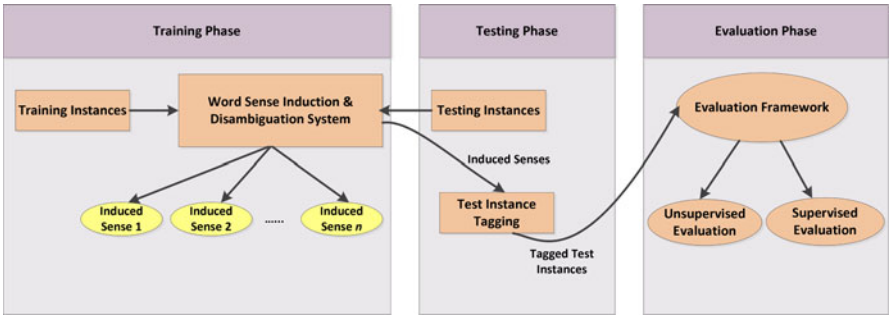

Fig. 2 Training, testing and evaluation phases of SemEval-2010 WSI Task

Table 6 Training and testing set details

	Training set	Testing set	Senses (#)
All	879,807	8,915	3.79
Nouns	716,945	5,285	4.46
Verbs	162,862	3,630	3.12

Table 7 Training set creation: example queries for target word *failure*

Word sense	Query
Sense 1	failure AND (loss OR nonconformity OR test OR surrender OR "force play" OR ...)
Sense 2	failure AND (ruination OR flop OR bust OR stall OR ruin OR walloping OR ...)

The created queries were issued to Yahoo! search API[5] and for each query a maximum of 1,000 pages were downloaded. For each page we extracted fragments of text that occurred in <p> < /p> html tags and contained the target word stem. In the final stage, each extracted fragment of text was POS-tagged using the Genia tagger (Tsuruoka and Tsujii 2005) and was only retained if the POS of the target word in the extracted text matched the POS of the target word in our dataset. The training dataset has been made available[6] to the research community.

3.2 Testing dataset

The testing dataset consisted of instances of the same target words given during the training phase. This dataset is part of the OntoNotes project (Hovy et al. 2006). The texts come from various news sources including CNN, ABC and others. For evaluation, we used the sense-tagged version of the dataset, in which target word

[5] http://developer.yahoo.com/search/ [Access:10/04/2010].

[6] http://www.cs.york.ac.uk/semeval2010_WSI/files/training_data.tar.gz.

instances are tagged with OntoNotes (Hovy et al. 2006) senses. The testing dataset has been made available[7] to the research community.

3.3 Participating systems

In this section we provide a brief description of the 26 systems (5 teams) that participated in the SemEval-2010 WSI task. Table 8 presents the key points of each method regarding their features and clustering method. Note that the symbols next to each system denote the corpus that was used to learn the senses of target words, i.e. * for the training corpus, + for the untagged version of the testing corpus and *+ for both.

Hermit Jurgens and Stevens (2010) presented a sense induction method that models the contexts of a target word in a high-dimensional word space using Random Indexing (RI) (Kanerva et al. 2000). RI represents the occurrence of a contextual word with a sparse index vector that is orthogonal to all other words index vectors with a high probability. A context of a target polysemous word is then represented by summing the index vectors corresponding to the n words occurring to the left and right of the target word. For clustering the target word contexts, they apply a hybrid method of K-Means and Hierarchical Agglomerative Clustering (HAC). Initially, context vectors are clustered using K-means, which assigns each context to its most similar cluster centroid. In the next step, the K induced clusters are repeatedly merged using HAC with average linkage. HAC stops cluster merging, when the two most similar clusters have a similarity less than a predefined threshold.

Duluth-WSI Pedersen (2010) participated in the WSI task with the SenseClusters (Purandare and Pedersen 2004; Pedersen 2007) WSI method. SenseClusters is a vector-based WSI system that constructs a word-by-word co-occurrence matrix by identifying bigrams or word co-occurrences (separated by up to n intervening words). Alternatively, the co-occurrence matrix can be constructed by considering unordered pairs of words. The co-occurrence matrix may be reduced to 300 dimensions by applying Singular Value Decomposition. The resulting co-occurrence matrix was exploited to create second order co-occurrence vectors each one representing a target word instance. Clustering of context vectors is performed by using the method of repeated bisections (rb) and the number of clusters, k, is automatically determined using either the PK2 measure or the Adapted Gap Statistic (Pedersen and Kulkarni 2006). The team submitted 16 runs, 5 out of which were random baselines.

UoY Korkontzelos and Manandhar (2010) presented a graph-based sense induction method. They initially construct a graph in which single nouns are represented as vertices. Subsequently, they generate noun pairs for each context of the target word and include them as vertices in the graph, if and only if these pairs are not distributionally similar to each one of their component nouns. Edges are drawn according to the distributional similarity of the corresponding vertices.

[7] http://www.cs.york.ac.uk/semeval2010_WSI/files/test_data.tar.gz.

Table 8 Participating systems overview

System	Features	Clustering method
KSU KDD (*)	String tokens	LDA + *K*-means
Hermit (*)	Word (pos + lemma)	*k*-means + HAC
UoY (*)	Word (pos + lemma), collocations	Chinese whispers
KCDC-GD (*)	Grammatical dependencies	Growing *k*-Means
KCDC-GD-2 (*)	Grammatical dependencies	Growing *k*-Means
KCDC-GDC (*)	Grammatical dependencies	Growing *k*-Means
KCDC-PC-2 (*)	Noun/verb phrases	Growing *k*-Means
KCDC-PC (*)	Distributionally expanded noun/verb phrases including the target word	Growing *k*-Means
KCDC-PT (*)	Noun/verb phrases including the target word	Growing *k*-Means
KCDC-PCGD (*)	Combination of KCDC-GD, KCDC-PC	Growing *k*-Means
Duluth-WSI (+)	Bigrams, ordered co-occurrences	Repeated bisections + PK2
Duluth-WSI-Gap (+)	Bigrams, ordered co-occurrences	Repeated bisections + GAP
Duluth-WSI-SVD (+)	Bigrams, ordered co-occurrences, SVD	Repeated bisections + PK2
Duluth-WSI-Co (+)	Unordered co-occurrences	Repeated bisections + PK2
Duluth-WSI-Co-Gap(+)	Unordered co-occurrences	Repeated bisections + GAP
Duluth-WSI-SVD-Gap(+)	Unordered co-occurrences, SVD	Repeated bisections + GAP
Duluth-Mix-Narrow-PK2 (*+)	Bigrams, ordered co-occurrences	Repeated bisections + PK2
Duluth-Mix-Narrow-Gap (*+)	Bigrams, ordered co-occurrences	Repeated bisections + GAP
Duluth-MIX-PK2 (*+)	Bigrams	Repeated bisections + PK2
Duluth-Mix-Gap (*+)	Bigrams	Repeated bisections + GAP
Duluth-Mix-Uni-PK2 (*+)	Unigrams	Repeated bisections + PK2
Duluth-Mix-Uni-Gap (*+)	Unigrams	Repeated bisections + GAP
Duluth-R-12 (+)	N/A	Random, 12 clusters
Duluth-R-13 (+)	N/A	Random, 13 clusters
Duluth-R-15 (+)	N/A	Random, 15 clusters
Duluth-R-110 (+)	N/A	Random, 110 clusters

Chinese Whispers (Biemann 2006) is applied to cluster the resulting graph. Each induced cluster is taken to represent one of the senses of the target word.

KCDC Kern et al. (2010) presented a sense induction method based on the vector-space model, which exploits a variety of grammatical and co-occurrence features. Specifically, each target word context was associated with a vector of features, i.e. grammatical dependencies, noun and verb phrases containing the target word, noun and verb phrases containing the target word that were also expanded with distributionally similar words and combinations of these features. Clustering of target word context vectors was performed using Growing *k*-Means (Daszykowski et al., 2002). The number of clusters *k* was automatically identified using a clustering evaluation stability criterion (Kern et al. 2010). The team submitted three runs to assess the influence of the random initialization of their clustering algorithm.

KSU KDD Elshamy et al. (2010) presented a sense induction based on LDA (Blei et al. 2003). In their model, the corpus of a target word consists of N contexts, where each one of them is represented by a multinomial distribution over C topics, which are in turn multinomial distributions over words. For each target polysemous word, Elshamy et al. (2010) trained a MALLET[8] parallel topic model implementation of LDA on all the training instances of that word. The trained topic model was then used to infer the topic distributions for each test instance of the target word. For a C-topics topic model, each topic distribution (for each test instance) was represented as a point in a C-dimensional topic space and K-means was then applied for clustering.

4 SemEval-2010 evaluation scheme

4.1 SemEval-2010 unsupervised evaluation

Following the SemEval-2007 WSI task (Agirre and Soroa 2007a), the SemEval-2010 WSI task also included an evaluation of WSI methods in a clustering task applying measures that intended to deal with the deficiencies of the previous competition as mentioned in Section 2.2.1. In SemEval-2010 WSI challenge there were two evaluation measures, i.e. *V-Measure* (Rosenberg and Hirschberg 2007) and (2) *paired F-Score* (Artiles et al. 2009). The implementations of *V-Measure* and *paired F-Score* have been made available[9] to the research community.

4.1.1 V-Measure

Let w be a target word with N instances (data points) in the testing dataset. Let $K = \{C_j | j = 1 \ldots n\}$ be a set of automatically generated clusters grouping these instances, and $S = \{G_i | i = 1 \ldots m\}$ the set of gold standard classes containing the desirable groupings of w instances.

V-Measure (Rosenberg and Hirschberg 2007) assesses the quality of a clustering solution by explicitly measuring its *homogeneity* and its *completeness*. Recall that homogeneity refers to the degree that each cluster consists of data points (target word instances) that primarily belong to a single gold standard class, while completeness refers to the degree that each gold standard class consists of data points primarily assigned to a single cluster (Rosenberg and Hirschberg 2007). Let h be homogeneity and c completeness. V-Measure is the harmonic mean of h and c, i.e. $VM = \frac{2 \cdot h \cdot c}{h+c}$.

Homogeneity The homogeneity, h, of a clustering solution is defined in Eq. 2, where $H(S|K)$ is the conditional entropy of the class distribution given the proposed clustering and $H(S)$ is the class entropy.

[8] http://mallet.cs.umass.edu.

[9] http://www.cs.york.ac.uk/semeval2010_WSI/files/evaluation.zip.

$$h = \begin{cases} 1, & \text{if } H(S) = 0 \\ 1 - \frac{H(S|K)}{H(S)}, & \text{otherwise} \end{cases} \tag{2}$$

$$H(S) = -\sum_{i=1}^{|S|} \frac{\sum_{j=1}^{|K|} a_{ij}}{N} \log \frac{\sum_{j=1}^{|K|} a_{ij}}{N} \tag{3}$$

$$H(S|K) = -\sum_{j=1}^{|K|} \sum_{i=1}^{|S|} \frac{a_{ij}}{N} \log \frac{a_{ij}}{\sum_{k=1}^{|S|} a_{kj}} \tag{4}$$

When $H(S|K)$ is 0, the solution is perfectly homogeneous, because each cluster only contains data points that belong to a single class. However in an imperfect situation, $H(S|K)$ depends on the size of the dataset and the distribution of class sizes. Hence, instead of taking the raw conditional entropy, V-Measure normalizes it by the maximum reduction in entropy the clustering information could provide, i.e. $H(S)$. When there is only a single class ($H(S) = 0$), any clustering would produce a perfectly homogeneous solution.

Completeness Symmetrically to homogeneity, the completeness, c, of a clustering solution is defined in Eq. 5, where $H(K|S)$ is the conditional entropy of the cluster distribution given the class distribution and $H(K)$ is the clustering entropy. When $H(K|S)$ is 0, the solution is perfectly complete, because all data points of a class belong to the same cluster.

Returning to our clustering example in Table 3, its V-Measure is equal to 0.275. In contrast, the V-Measure of the clustering solution in Table 4 is 0.45. This result shows that V-measure is able to discriminate between these two clusterings in terms of homogeneity and completeness by considering the make-up of the clusters beyond the majority class. The ability of V-Measure to discriminate between two clusterings, when one of them has a better homogeneity (resp. completeness) has also been observed in (Amigó et al. 2009).

$$c = \begin{cases} 1, & \text{if } H(K) = 0 \\ 1 - \frac{H(K|S)}{H(K)}, & \text{otherwise} \end{cases} \tag{5}$$

$$H(K) = -\sum_{j=1}^{|K|} \frac{\sum_{i=1}^{|S|} a_{ij}}{N} \log \frac{\sum_{i=1}^{|S|} a_{ij}}{N} \tag{6}$$

$$H(K|S) = -\sum_{i=1}^{|S|} \sum_{j=1}^{|K|} \frac{a_{ij}}{N} \log \frac{a_{ij}}{\sum_{k=1}^{|K|} a_{ik}} \tag{7}$$

4.1.2 Paired F-Score

In this evaluation, the clustering problem is transformed into a classification problem of pairs of target word instances. For each cluster C_i, one can generate $\binom{|C_i|}{2}$

instance pairs, where $|C_i|$ is the total number of instances that have been tagged with cluster C_i. Similarly, for each gold standard class G_i one can generate $\binom{|G_i|}{2}$ instance pairs, where $|G_i|$ is the total number of instances that belong to gold standard class G_i.

Let $F(K)$ be the set of instance pairs that exist in the automatically induced clusters and $F(S)$ be the set of instance pairs that exist in the gold standard. Precision is the ratio of the number of common instance pairs between the two sets to the total number of pairs in the clustering solution (Eq. 8), while recall is the ratio of the number of common instance pairs between the two sets to the total number of pairs in the gold standard (Eq. 9). Finally, precision and recall are combined to produce the harmonic mean ($FS = \frac{2 \cdot P \cdot R}{P+R}$).

$$P = \frac{|F(K) \cap F(S)|}{|F(K)|} \tag{8}$$

$$R = \frac{|F(K) \cap F(S)|}{|F(S)|} \tag{9}$$

For example in Table 3, the paired F-Score for that clustering solution is equal to 0.55. In contrast, for the clustering solution in Table 4 the paired F-Score is equal to 0.59.

4.1.3 Results and discussion

In this section, we present the results of the top 10 best performing systems in the unsupervised evaluation along with three baselines. The first baseline, Most Frequent Sense (*MFS*), groups all testing instances of a target word into one cluster. Note that the *MFS* baseline is equivalent to the *1Cl1W* baseline that was used in the SemEval-2007 WSI task (Agirre and Soroa 2007a). The second baseline, *Random*, randomly assigns an instance to one out of four clusters. The number of clusters of *Random* was chosen to be roughly equal to the average number of senses in the GS. This baseline is executed five times and the results are averaged. The *1Cl1Ins* baseline creates a cluster for each instance of a target word.

Table 9 shows the top 10 best performing systems using the first evaluation measure. The last column shows the number of induced clusters of each system in the test set. The V-Measure of the *MFS* is by definition equal to 0. Since this baseline groups all instances of a target word into a single cluster, its completeness is 1 and its homogeneity is 0.

As can be observed, all participating systems outperform the *MFS* baseline, apart from one. Regarding the *Random* baseline, we observe that 17 systems perform better, which shows that they have learned useful information better than chance.

Table 9 also shows that V-Measure tends to favor systems producing a higher number of clusters than the number of GS senses. For instance, the *1Cl1Ins* baseline produces an average of 89.15 clusters per target word and has achieved the highest

Table 9 V-Measure unsupervised evaluation

System	VM (%) (all)	VM (%) (nouns)	VM (%) (verbs)	#Cl
1Cl1Ins	31.7	25.6	35.8	89.15
Hermit	16.2	16.7	15.6	10.78
UoY	15.7	20.6	8.5	11.54
KSU KDD	15.7	18	12.4	17.5
Duluth-WSI	9	11.4	5.7	4.15
Duluth-WSI-SVD	9	11.4	5.7	4.15
Duluth-R-110	8.6	8.6	8.5	9.71
Duluth-WSI-Co	7.9	9.2	6	2.49
KCDC-PCGD	7.8	7.3	8.4	2.9
KCDC-PC	7.5	7.7	7.3	2.92
KCDC-PC-2	7.1	7.7	6.1	2.93
Random	4.4	4.2	4.6	4
MFS	0	0	0	1

V-Measure that no system managed to outperform. The homogeneity of that baseline is equal to 1, since each cluster contains one and only one instance of a gold standard class. The completeness, however, of that baseline is not 0, as one might expect, since each cluster captures a small amount (one instance) of the total number instances of a gold standard class. Hence, the harmonic mean of homogeneity and completeness for that baseline achieve a score which seems to be high compared to systems participating in the task.

The bias of V-Measure towards clustering solutions with a large number of clusters motivated us to introduce the second unsupervised evaluation measure (paired F-Score) that penalizes systems when they produce: (1) a higher number of clusters (low recall) or (2) a lower number of clusters (low precision), than the gold standard number of senses.

Table 10 shows the top 10 best performing systems using the second unsupervised evaluation measure. In this evaluation we again observe that most of the systems perform better than *Random*. All systems perform better than the *1Cl1Ins* baseline which achieves the lowest paired F-Score due to its very low recall. Despite that, we also observe that no system performs better than the *MFS* baseline. In fact, it appears that the relationship between V-Measure and paired F-Score is inversely predictive. The *MFS* achieves a higher paired F-Score compared to the rest of the systems, because its recall is always 1, while its precision is well above 0, due to the dominance of the *MFS* in the dataset. Specifically, in skewed sense distributions most target word instance pairs on the gold standard are generated from the *MFS*, which in effect allows that baseline to achieve a moderate precision.

Additionally, it seems that systems generating a smaller number of clusters than the GS number of senses are biased towards the *MFS*, hence they are not able to perform better. On the other hand, systems generating a higher number of clusters are penalized by this measure (low recall), while systems generating a number of

Table 10 Paired F-Score unsupervised evaluation

System	FS (%) (all)	FS (%) (nouns)	FS (%) (verbs)	#Cl
MFS	63.5	57.0	72.7	1
Duluth-WSI-SVD-Gap	63.3	57.0	72.4	1.02
KCDC-PT	61.8	56.4	69.7	1.5
KCDC-GD	59.2	51.6	70.0	2.78
Duluth-Mix-Gap	59.1	54.5	65.8	1.61
Duluth-Mix-Uni-Gap	58.7	57.0	61.2	1.39
KCDC-GD-2	58.2	50.4	69.3	2.82
KCDC-GDC	57.3	48.5	70.0	2.83
Duluth-Mix-Uni-PK2	56.6	57.1	55.9	2.04
KCDC-PC	55.5	50.4	62.9	2.92
KCDC-PC-2	54.7	49.7	61.7	2.93
Random	31.9	30.4	34.1	4
1Cl1Ins	0.09	0.08	0.11	89.15

clusters roughly the same as the number of gold standard senses tend to conflate these senses a lot more than the *MFS*.

4.2 Semeval-2010 supervised evaluation

In this evaluation, the testing dataset is split into a mapping and an evaluation corpus. The first one is used to map the automatically induced clusters to gold standard senses, while the second is used to evaluate methods in a WSD setting. This evaluation follows the supervised evaluation of SemEval-2007 WSI task Agirre and Soroa (2007b) described in Section 2.2.2, with the difference that the reported results are an average of 5 random splits. This repeated random sampling was performed to overcome the deficiencies of the SemEval-2007 WSI challenge, in which different splits were providing different system rankings. The supervised evaluation scripts and dataset split has been made available[10] to the research community.

4.2.1 Results and discussion

In this section we present the results of the 26 systems along with two baselines, i.e. *MFS* and *Random*. Note that the *1Cl1Ins* baseline is not defined in this evaluation setting, since clusters appearing in the mapping corpus do not appear in the evaluation corpus and the mapping cannot be performed.

Table 11 shows the results of this evaluation for a 80–20 test set split, i.e. 80 % for mapping and 20 % for evaluation, for the top 10 best performing systems. The last column shows the average number of gold standard senses identified by each system in

[10] http://www.cs.york.ac.uk/semeval2010_WSI/files/evaluation.zip.

Table 11 Supervised recall (SR) (test set split:80 % mapping, 20 % evaluation)

System	SR (%) (all)	SR (%) (nouns)	SR (%) (verbs)	#S
UoY	62.4	59.4	66.8	1.51
Duluth-WSI	60.5	54.7	68.9	1.66
Duluth-WSI-SVD	60.5	54.7	68.9	1.66
Duluth-WSI-Co-Gap	60.3	54.1	68.6	1.19
Duluth-WSI-Co	60.8	54.7	67.6	1.51
Duluth-WSI-Gap	59.8	54.4	67.8	1.11
KCDC-PC-2	59.8	54.1	68.0	1.21
KCDC-PC	59.7	54.6	67.3	1.39
KCDC-PCGD	59.5	53.3	68.6	1.47
KCDC-GDC	59.1	53.4	67.4	1.34
MFS	58.7	53.2	66.6	1
Random	57.3	51.5	65.7	1.53

the five splits of the evaluation datasets. In this evaluation setting, 14 systems perform better than the *MFS* baseline and 17 perform better than *Random*. The ranking of systems with respect to the part-of-speech of the target word is different, which in effect indicates that the two POS classes should be treated differently by WSI methods in terms of the clustering algorithm, features and parameters tuning.

As it has already been mentioned, the supervised evaluation changes the distribution of clusters by mapping each cluster to a weighted vector of senses. As a result, it has the tendency to favor systems generating a higher number of clusters depending on the homogeneity of the corresponding clusters. For that reason, we applied a second testing set split, where we decreased the size of the mapping corpus (60 %) and increased the size of the evaluation corpus (40 %). The reduction of the mapping corpus size allows us to observe, whether the above statement is correct, since systems with a high number of clusters could potentially suffer from an unreliable mapping of their induced clusters to gold standard senses.

Table 12 shows the results of the second supervised evaluation. The ranking of participants did not change significantly, i.e. we observe only different rankings among systems belonging to the same participant. Despite that, Table 12 also shows that the reduction of the mapping corpus has a different impact on systems generating a larger number of clusters than the gold standard number of senses.

For instance, *UoY* that generated 11.54 clusters tends to perform similarly in both splits with respect to its distance from the *MFS*. The reduction of the mapping size did not have any significant impact. In contrast, *KSU KDD* that generates 17.5 clusters was below the *MFS* by 6.49 % in the 80–20 split and by 7.83 % in the 60–40 split. We observe that the reduction of the mapping corpus had a negative impact in this case. The overall conclusion is that systems generating a skewed distribution, in which a small number of homogeneous clusters tag the majority of instances and a larger number of clusters tag only a few instances, are likely to have a better performance than systems that produce a more uniform distribution in this dataset.

Table 12 Supervised recall (SR) (test set split:60 % mapping, 40 % evaluation)

System	SR (%) (All)	SR (%) (Nouns)	SR (%) (Verbs)	#S
UoY	62.0	58.6	66.8	1.66
Duluth-WSI-Co	60.1	54.6	68.1	1.56
Duluth-WSI-Co-Gap	59.5	53.5	68.3	1.2
Duluth-WSI-SVD	59.5	53.5	68.3	1.73
Duluth-WSI	59.5	53.5	68.3	1.73
Duluth-WSI-Gap	59.3	53.2	68.2	1.11
KCDC-PCGD	59.1	52.6	68.6	1.54
KCDC-PC-2	58.9	53.4	67.0	1.25
KCDC-PC	58.9	53.6	66.6	1.44
KCDC-GDC	58.3	52.1	67.3	1.41
MFS	58.3	52.5	66.7	1
Random	56.5	50.2	65.7	1.65

5 Evaluation according to the skewness of the distribution of senses

Both the Semeval-2007 and SemEval-2010 WSI tasks have evaluated sense induction methods on two classes of words, i.e. nouns and verbs. Therefore, both evaluation schemes have ignored an important aspect of Word Sense Induction and Disambiguation, i.e. the skewness of the target word distribution of senses. A contrastive evaluation according to the skewness of sense distribution would possibly shed light on how different features and clustering methods perform under highly skewed, less skewed or even uniform distribution of senses.

Véronis (2004) had criticized vector-based methods as being unable to detect rare senses of words and suggested a graph-based clustering method that was able to detect senses whose relative frequency was more than 5 %. In particular, Véronis (2004) showed that the attempts to replicate the results of Schütze (1998) only succeeded when the actual senses were few in number, more or less equiprobable and highly individualized.

In this section, we evaluate the SemEval-2010 WSI participating methods in both unsupervised and supervised evaluation settings by dividing the target words into three categories according to the skewness of their distribution of senses. Equation 10 defines the skewness of a distribution, where x_i refers to the frequency of sense i, i.e. number of target word instances that have been tagged with sense i in the gold standard, $\bar{x}$ refers to the mean of the distribution and N is the total number of target word instances.

$$G = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^3}{\left(\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2\right)^{\frac{3}{2}}} \tag{10}$$

Table 13 provides a description of the three categories that we generated in terms of skewness range for each category, the total number of instances and the average

Table 13 Statistics of skewness categories

Category	Instances	Nouns		Verbs		All
		Skewness	Senses	Skewness	Senses	Senses
(1)	2949	0.01–0.53	4.1	0.0–0.56	3.3	3.6
(2)	2851	0.55–0.88	3.8	0.57–0.71	2.5	3.1
(3)	3115	0.88–1.71	5.7	0.73–1.15	4.2	5.1

Table 14 V-Measure unsupervised evaluation in the three skewness categories

Skewness category					
(1)		(2)		(3)	
System CL	VM (%)	System	VM (%)	System	VM (%)
1Cl1Ins	37.5	*1Cl1Ins*	28.7	*1Cl1Ins*	28.9
KSU KDD	20.0	UoY	15.3	UoY	16.3
Hermit	19.5	Hermit	14.6	Hermit	14.6
UoY	15.3	KSU KDD	13.7	KSU KDD	14.0
KCDC-PCGD	11.5	Duluth-WSI	9.9	Duluth-WSI	8.0
Duluth-R-110	10.3	Duluth-WSI-SVD	9.9	Duluth-WSI-SVD	8.0
Random	5.3	*Random*	3.7	*Random*	4.2

Top five participating systems are shown

number of senses for each POS class. For a given POS class (noun or verb) the three categories were generated by following the following process:

1. The skewness of target words was calculated.
2. Target words were sorted according to their skewness
3. All target words were assigned to one skewness category, so that all three categories roughly have the same total number of target word instances.

5.1 Unsupervised evaluation

5.1.1 Results using V-Measure

Table 14 shows the V-Measure performance of the top five participating systems and baselines in the three skewness categories. In all categories, we observe that none of the systems was able to perform better than the *1Cl1Ins* baseline, while most of the systems were able to perform better than *Random*. As in the official evaluation, we also observe that systems generating a higher number of clusters[11] achieve a high V-Measure, although their performance does not increase

[11] The number of clusters of each system is shown in Table 9.

monotonically with the number of clusters increasing. Recall that all systems perform better than the *MFS*, since its V-Measure is 0.

By comparing the ranking of systems in the second and third skewness categories of Table 14 we do not observe any difference. Despite that, the ranking is different in the first and second skewness categories, as well as in the first and third. For instance, *KCDC-PCGD* that was ranked 13th in the official evaluation, performs significantly better in the first skewness category despite the small number of generated clusters. This result indicates that the particular system tends to perform better when sense distributions tend to be equiprobable, and worse when moving on to more skewed distributions.

In contrast, systems *Duluth-WSI* and *Duluth-WSI-SVD*, which perform well in the second and third skewness categories, are not included in the top five systems of the first category. This result indicates that these systems perform better in more skewed distributions.

5.1.2 Results using paired F-Score

Table 15 shows the paired F-Score performance of the top five participating systems and baselines in the three skewness categories. In all categories, we observe that no system was able to perform better than the *MFS* baseline, while most of the systems perform better than *Random*. As the official evaluation has shown, systems generating a very small number of clusters (see footnote 11) tend to be biased towards the *MFS* baseline and achieved a high paired F-Score.

By comparing the ranking of systems in the three skewness categories of Table 15 we do not observe any significant differences. Specifically, *Duluth-WSI-SVD-Gap* and *KCDC-PT* perform in most categories better than other systems as a result of the their small number of clusters. Given that performance in the paired F-Score seems to be more biased towards a small number of clusters, than V-Measure was towards a high number of clusters, the particular evaluation measure does not offer any discriminative information among the three categories.

Table 15 Paired F-Score (FS) unsupervised evaluation in the three skewness categories

Skewness category					
(1)		(2)		(3)	
System	FS (%)	System	FS (%)	System	FS (%)
MFS	56.5	*MFS*	66.5	*MFS*	67.2
Duluth-WSI-SVD-GAP	56.5	Duluth-WSI-SVD-GAP	66.1	Duluth-WSI-SVD-GAP	67.2
KCDC-PT	55.9	KCDC-PT	64.4	KCDC-PT	65.1
Duluth-Mix-Uni-Gap	53.8	Duluth-Mix-Uni-Gap	63.4	KCDC-GD	64.4
Duluth-Mix-Gap	53.7	KCDC-GD-2	61.4	Duluth-Mix-Gap	63.1
KCDC-GD-2	52.9	KCDC-GDC	61.4	KCDC-GD-2	60.2
Random	30.1	*Random*	32.7	*Random*	33.1
1Cl1Ins	0.1	*1Cl1Ins*	0.1	*1Cl1Ins*	0.1

Top five participating systems are shown

Table 16 Supervised recall (SR) (test set split: 80 % mapping, 20 % evaluation) in the three skewness categories

Skewness category					
(1)		(2)		(3)	
System	SR (%)	System	SR (%)	System	SR (%)
UoY	51.9	UoY	65.7	UoY	69.9
Duluth-Mix-Narrow-Gap	51.4	Duluth-WSI-SVD	65.4	KCDC-PC	66.4
Hermit	51.2	Duluth-WSI	65.4	KCDC-PC-2	66.4
KCDC-PCGD	51.0	Duluth-WSI-Co-Gap	64.9	KCDC-PT	66.3
Duluth-Mix-Narrow-PK2	50.9	KCDC-PC	64.5	Duluth-WSI-Co-Gap	66.2
Duluth-WSI-SVD	50.6	Duluth-WSI-Co	64.5	Duluth-WSI-Co	66.1
Duluth-WSI	50.6	Duluth-WSI-Gap	64.3	*MFS*	65.9
Duluth-WSI-Co	50.5	KCDC-PC-2	63.5	Random	65.0
Duluth-WSI-Co-Gap	50.3	KCDC-GDC	63.0		
KCDC-GD	49.8	KCDC-GD-2	62.4		
KCDC-PC-2	49.7	Hermit	62.4		
Duluth-WSI-Gap	49.5	Duluth-WSI-SVD-Gap	62.1		
Duluth-R-13	49.3	*MFS*	62.1		
KCDC-GDC	50.0	Random	61.0		
KCDC-GD-2	48.7				
KCDC-PT	48.6				
KCDC-PC	48.5				
MFS	48.1				
Random	45.9				

Only systems performing better than the *MFS* are shown

5.2 Supervised evaluation

Table 16 shows the supervised recall of participating systems that managed to perform better than the *MFS* in the 80–20 split of the dataset. As can be observed, in the first skewness category in which the distributions of target word senses are less skewed, 17 systems managed to outperform the *MFS*, where in most cases the performance differences are statistically significant (McNemar's test, 95 % confidence level).

Despite that, as we move to the second and third skewness categories in which the distributions of word senses become more and more skewed, we observe that a decreasing number of systems performs better than the *MFS*. Specifically, in the second skewness category 12 systems managed to perform better than the *MFS*. In the third skewness category, this picture becomes worse since only six systems outperformed this baseline. Overall, it becomes apparent that the majority of sense induction systems perform worse as word sense distributions become more skewed.

For instance in Table 16, we observe that *Hermit* performs well in the first skewness category (its position in the official evaluation was 17th) outperforming

the *MFS* by 3.08 %. In the second category, *Hermit* outperforms the *MFS* by *0.21 %*, while in the third category it performs worse than the *MFS*.

Figure 3 shows the performance differences from the *MFS* for all systems that perform better than this baseline in all skewness categories. As can be observed, the performance difference of all systems, apart from *KCDC-PC* and *UoY*, decreases as skewness increases. Interestingly *KCDC-PC* performs better in the second skewness category, while *UoY* is the only system whose performance difference from the *MFS* remains roughly the same along the three categories. Specifically, *UoY* outperforms the *MFS* by 3.72 % in the first, 3.56 % in the second and 4 % in the third category.

5.3 Further discussion

Given that one of the primary aims of WSI is to build better sense inventories, it would be interesting to re-visit the method of the system that performs consistently above the *MFS* baseline as skewness increases, and draw conclusions useful for lexicographers and linguists.

UoY (Korkontzelos and Manandhar 2010) is a graph-based method, in which each vertex corresponds to either a single noun, or a pair of nouns co-occurring with the target word. A single noun vertex is generated when the noun is judged to be unambiguous, i.e. it appears with only one sense of the target word. Otherwise, the noun is taken to be ambiguous and is combined with any other unambiguous noun to form a pair. The method of determining whether a noun is ambiguous or not is described in detail in Korkontzelos and Manandhar (2010).

In the next step, hard clustering of the constructed graph generates the clusters (senses) and allows one ambiguous noun to be part of more than one clusters (senses) of the target word by participating in more than one noun-noun pairs.

It appears that soft clustering methods that attempt to reduce the ambiguity of the extracted features through the use of collocations (as in *UoY*) can produce less-sense conflating clusters. These induced clusters correspond both to frequent and rare senses of words, hence the output of such methods could be exploited by

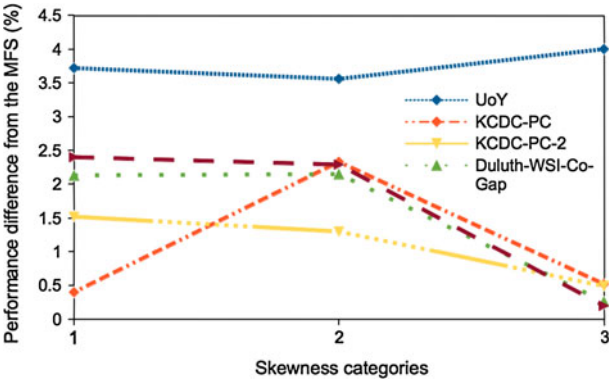

Fig. 3 Performance difference from the *MFS* for 5 systems

Table 17 BCubed unsupervised evaluation

System	BCubed (%) (all)	BCubed (%) (nouns)	BCubed (%) (verbs)	#Cl
MFS	64.1	57.6	73.4	1
Duluth-WSI-SVD-Gap	64.0	57.6	73.2	1.02
KCDC-PT	63.1	57.4	71.2	1.5
KCDC-GD	61.2	53.9	71.8	2.78
KCDC-GD-2	60.5	53.1	71.3	2.82
Duluth-Mix-Gap	60.5	56.0	67.2	1.61
Duluth-Mix-Uni-Gap	59.7	57.6	62.6	1.39
KCDC-GDC	59.4	50.8	71.9	2.83
Duluth-Mix-Uni-PK2	57.9	57.8	58.1	2.04
KCDC-PC	57.6	52.4	65.3	2.92
KCDC-PC-2	57.0	52.0	64.3	2.93
Random	35.2	33.4	37.7	4
1Cl1Ins	8.0	7.9	8.2	89.15

Top ten participating systems are shown

lexicographers as additional assistance in their hard-task of identifying infrequent or idiomatic senses of words.

6 Conclusion and future work

This paper presented a comprehensive description of the SemEval-2010 word sense induction challenge focusing on the task description, resources used, participating systems, evaluation framework, as well as the main differences of the task from the corresponding SemEval-2007 WSI challenge. Subsequently, we evaluated participating systems in terms of their unsupervised (V-Measure, paired F-Score) and supervised (supervised recall) performance according to the skewness of target words distribution of senses.

The evaluation has shown that the current state-of-the-art lacks unbiased measures that objectively evaluate the clustering solutions of sense induction systems. Recently, Amigó et al. (2009) showed that BCubed (Bagga and Baldwin 1998) is a less biased measure than entropy-based ones (e.g. V-Measure) or measures based on counting pairs (e.g. paired F-Score), since it is able to satisfy a set of mathematical constraints mentioned in Amigó et al. (2009) that others do not.

BCubed decomposes the evaluation process by: (1) evaluating the precision and recall of each data point, (2) averaging the calculated figures, and (3) producing the harmonic mean of the averaged precision and recall. The precision of a data point x represents how many other data points in the same cluster belong to the same gold standard class as x, while recall represents how many data points from the class of x belong to the same cluster as x. A data point with high BCubed recall means that we would find most of its related data points without leaving the cluster (Amigó

et al. 2009). Similarly, high precision means that we would not find noisy points in the same cluster (Amigó et al., 2009).

In contrast to V-Measure that evaluates each cluster (resp. each class), BCubed recall and precision are computed over single data points, in effect being less biased towards the predominant class. Compared to paired F-Score, BCubed's computation over single data points reduces the quadratic effect caused by the cluster size (Amigó et al. 2009). Despite that, our experiments on evaluating sense induction methods using BCubed showed a very high correlation with the ranking of systems as produced by paired F-Score. Table 17 shows the top 10 best performing systems using the BCubed measure. As can be observed the ranking is identical to the paired F-Score ranking (Table 10).

Based on our current results, it seems that the assessment on a task-oriented basis is more appropriate allowing one to identify which features or clustering methods benefit which applications. Given that different applications or domains may require different sense granularity, such evaluations would possibly enhance our understanding of computational semantics and extend the current state-of-the-art, provided that they correspond to clearly-defined end-user applications.

The second evaluation scheme, i.e. supervised evaluation, could be considered as a task oriented-application, since it transforms WSI systems to semi-supervised WSD ones. Therefore, we believe that it is a useful evaluation setting, in which the results of systems can be interpreted in terms of the number of generated clusters and the distribution of target word instances within the clusters. Moreover, Navigli and Crisafulli (2010) have a presented an application of sense induction to web search result clustering and showed that the use of WSI improves the quality of search result clustering and enhances the diversification of search results. This is another application-oriented evaluation that could be explored in the future.

Another angle for evaluating WSI methods could focus on two important factors affecting their performance. The first one is the skewness of the distribution of gold standard senses, and the second is the similarity between gold standard senses. For the first factor, we presented an evaluation setting in which we split the dataset in three skewness categories and showed that the ranking of systems (especially in the supervised evaluation) changes with respect to the level of skewness. For the second factor, one could measure sense similarity in different ways (e.g. in a distributional similarity framework or by exploiting WordNet-type similarity measures such as *Jiang-Conrath* similarity (Jiang and Conrath 1997)), and then assess WSI systems on their ability to distinguish senses with different levels of similarity.

Acknowledgments We gratefully acknowledge the support of the EU FP7 INDECT project, Grant No. 218086, the National Science Foundation Grant NSF-0715078, Consistent Criteria for Word Sense Disambiguation, and the GALE program of the Defense Advanced Research Projects Agency, Contract No. HR0011-06-C-0022, a subcontract from the BBN-AGILE Team.

References

Agirre, E., Ansa, O., Hovy, E., & Martinez, D. (2001). Enriching wordnet concepts with topic signatures. *ArXiv Computer Science e-prints*.

Agirre, E., & De Lacalle, O. L. (2003). Clustering wordnet word senses. In *Proceedings of the conference on recent advances on natural language (RANLP'03)*, Borovets, Bulgaria.

Agirre, E., & De Lacalle, O. L. (2004). Publicly available topic signatures for all wordnet nominal senses. In *Proceedings of the 4th international conference on language resources and evaluation(LREC)*, Lisbon, Portugal.

Agirre, E., Martínez, D., de Lacalle, O. L., & Soroa, A. (2006a). Evaluating and optimizing the parameters of an unsupervised graph-based wsd algorithm. In *Proceedings of the first workshop on graph based methods for natural language processing*, TextGraphs-1 (pp. 89–96). Stroudsburg, PA, USA: Association for Computational Linguistics.

Agirre, E., Martínez, D., López de Lacalle, O., & Soroa, A. (2006b). Two graph-based algorithms for state-of-the-art wsd. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 585–593). Sydney, Australia: ACL.

Agirre, E., & Soroa, A. (2007a). Semeval-2007 task 02: Evaluating word sense induction and discrimination systems. In *Proceedings of the fourth international workshop on semantic evaluations* (pp. 7–12). Prague, Czech Republic: ACL.

Agirre, E., & Soroa, A. (2007b). Ubc-as: A graph based unsupervised system for induction and classification. In *Proceedings of the fourth international workshop on semantic evaluations (SemEval-2007)* (pp. 346–349). Prague, Czech Republic: Association for Computational Linguistics.

Alfonseca, E., & Manandhar, S. (2002). Extending a lexical ontology by a combination of distributional semantics signatures. In *Proceedings of the 13th international conference on knowledge engineering and knowledge management. Ontologies and the semantic web*, EKAW '02 (pp. 1–7). London, UK: Springer.

Amigó, E., Gonzalo, J., Artiles, J., & Verdejo, F. (2009). A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Inf. Retr., 12*, 461–486.

Artiles, J., Amigó, E., & Gonzalo, J. (2009). The role of named entities in Web People Search. In *Proceedings of the 2009 conference on empirical methods in natural language processing* (pp. 534–542). Singapore: Association for Computational Linguistics.

Bagga, A., & Baldwin, B. (1998). Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 36th annual meeting of the association for computational linguistics and 17th international conference on computational linguistics—Volume 1*, ACL '98 (pp. 79–85). Stroudsburg, PA, USA: Association for Computational Linguistics.

Baker, C. F., Fillmore, C. J., & Lowe, J. B. (1998). The berkeley framenet project. In *Proceedings of the 36th annual meeting of the association for computational linguistics and 17th international conference on computational linguistics—Volume 1*, ACL '98 (pp. 86–90). Stroudsburg, PA, USA: Association for Computational Linguistics.

Biemann, C. (2006). Chinese whispers—An efficient graph clustering algorithm and its application to natural language processing problems. In *Proceedings of textGraphs* (pp. 73–80). New York,USA: ACL.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *J. Mach. Learn. Res., 3*, 993–1022.

Brody, S., & Lapata, M. (2009). Bayesian word sense induction. In *Proceedings of the 12th conference of the european chapter of the association for computational linguistics*, EACL '09 (pp. 103–111). Stroudsburg, PA, USA: Association for Computational Linguistics.

Clauset, A., Moore, C., & Newman, M. E. J. (2008). Hierarchical structure and the prediction of missing links in networks. *Nature, 453*(7191), 98–101.

Daszykowski, M., Walczak, B., & Massart, D. L. (2002). On the optimal partitioning of data with k-means, growing k-means, neural gas, and growing neural gas. *Journal of Chemical Information and Computer Sciences, 42*(6), 1378–1389.

Dorow, B., & Widdows, D. (2003). Discovering corpus-specific word senses. In *Proceedings of the 10th conference of the European chapter of the ACL* (pp. 79–82). Budapest, Hungary: ACL.

Elshamy, W., Caragea, D., & Hsu, W. (2010). Ksu kdd: Word sense induction by clustering in topic space. In *Proceedings of the 5th international workshop on semantic evaluation* (pp. 367–370). Uppsala, Sweden: Association for Computational Linguistics.

Fellbaum, C. (1998). *Wordnet: An Electronic Lexical Database*. Cambridge, MA, USA: MIT Press.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., & Weischedel, R. (2006). Ontonotes: The 90 % solution. In *Proceedings of the human language technology / North American Association for computational linguistics conference*, pp. 57–60. New York, USA.

Jiang, J. J., & Conrath, D. W. (1997). Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In *International conference research on computational linguistics*, pp. 19–33.

Jurgens, D., & Stevens, K. (2010). Hermit: Flexible clustering for the semeval-2 wsi task. In *Proceedings of the 5th international workshop on semantic evaluation* (pp. 359–362). Uppsala, Sweden: Association for Computational Linguistics.

Kanerva, P., Kristoferson, J., & Anders, H. (2000). Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd annual conference of the cognitive science society* (pp. 10–36). Uppsala, Sweden.

Kern, R., Muhr, M., & Granitzer, M. (2010). Kcdc: Word sense induction by using grammatical dependencies and sentence phrase structure. In *Proceedings of the 5th international workshop on semantic evaluation* (pp. 351–354). Uppsala, Sweden: Association for Computational Linguistics.

Kilgarriff, A., Kovář, V., Krek, S., Srdanović, I., & Tiberius, C. (2010). A quantitative evaluation of word sketches. In *Proceedings of the XIV Euralex international Congress*, pp. 251–263, Leeuwarden, Netherlands. Leeuwarden: Fryske Academy.

Klapaftis, I., & Manandhar, S. (2010). Word sense induction & disambiguation using hierarchical random graphs. In *Proceedings of the 2010 conference on empirical methods in natural language processing* (pp. 745–755). Cambridge, MA: Association for Computational Linguistics.

Korkontzelos, I., & Manandhar, S. (2010). Uoy: Graphs of unambiguous vertices for word sense induction and disambiguation. In *Proceedings of the 5th international workshop on semantic evaluation* (pp. 355–358). Uppsala, Sweden: Association for Computational Linguistics.

Lin, D., & Pantel, P. (2002). Concept discovery from text. In *Proceedings of the 19th international conference on computational linguistics* (pp. 1–7). Morristown, NJ, USA: Association for Computational Linguistics.

Manandhar, S., Klapaftis, I., Dligach, D., & Pradhan, S. (2010). Semeval-2010 task 14: Word sense induction & disambiguation. In *Proceedings of the 5th international workshop on semantic evaluation* (pp. 63–68). Uppsala, Sweden: Association for Computational Linguistics.

Navigli, R., & Crisafulli, G. (2010). Inducing word senses to improve web search result clustering. In *Proceedings of the 2010 conference on empirical methods in natural language processing* (pp. 116–126). Cambridge, MA: Association for Computational Linguistics.

Niu, Z.-Y., Ji, D.-H., & Tan, C.-L. (2007). I2r: Three systems for word sense discrimination, chinese word sense disambiguation, and english word sense disambiguation. In *Proceedings of the fourth international workshop on semantic evaluations (SemEval-2007)* (pp. 177–182). Prague, Czech Republic: Association for Computational Linguistics.

Pedersen, T. (2007). Umnd2: Senseclusters applied to the sense induction task of senseval-4. In *Proceedings of the fourth international workshop on semantic evaluations* (pp. 394–397). Prague, Czech Republic: ACL.

Pedersen, T. (2010). Duluth-wsi: Senseclusters applied to the sense induction task of semeval-2. In *Proceedings of the 5th international workshop on semantic evaluation* (pp. 363–366). Uppsala, Sweden: Association for Computational Linguistics.

Pedersen, T., & Kulkarni, A. (2006). Automatic cluster stopping with criterion functions and the gap statistic. In *Proceedings of the 2006 conference of the North American chapter of the ACL on human language technology* (pp. 276–279). Morristown, NJ, USA: ACL.

Pinto, D., Rosso, P., & Jiménez-Salazar, H. (2007). Upv-si: Word sense induction using self term expansion. In *Proceedings of the fourth international workshop on semantic evaluations (SemEval-2007)* (pp. 430–433). Prague, Czech Republic: Association for Computational Linguistics.

Purandare, A., & Pedersen, T. (2004). Senseclusters - finding clusters that represent word senses. In D. M. Susan Dumais & S. Roukos (Eds.), *HLT-NAACL 2004: Demonstration Papers*, (pp. 26–29). Boston, USA: ACL.

Rosenberg, A., & Hirschberg, J. (2007). V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)* (pp. 410–420). Prague, Czech Republic.

Salton, G., & Buckley, C. (1988). Term weighting approaches in automatic text retrieval. *Information Processing and Management, 24*(5), 513–523.

Schütze, H. (1998). Automatic word sense discrimination. *Computational Linguistics, 24*(1), 97–123.

Tsuruoka, Y., & Tsujii, J. (2005). Bidirectional inference with the easiest-first strategy for tagging sequence data. In *HLT '05: Proceedings of the conference on human language technology and*

empirical methods in natural language processing (pp. 467–474). Morristown, NJ, USA: Association for Computational Linguistics.
Véronis, J. (2004). Hyperlex: lexical cartography for information retrieval. *Computer Speech & Language, 18*(3), 223–252.