# Neural context embeddings for automatic discovery of word senses

**Mikael Kågebäck, Fredrik Johansson, Richard Johansson†, Devdatt Dubhashi**
Computer Science & Engineering, Chalmers University of Technology
SE-412 96, Göteborg, Sweden
†Språkbanken, University of Gothenburg
Box 200, Göteborg, Sweden
{kageback,frejohk,dubhashi}@chalmers.se
richard.johansson@svenska.gu.se

## Abstract

*Word sense induction* (WSI) is the problem of automatically building an inventory of senses for a set of target words using only a text corpus. We introduce a new method for embedding word instances and their context, for use in WSI. The method, *Instance-context embedding* (ICE), leverages neural word embeddings, and the correlation statistics they capture, to compute high quality embeddings of word contexts. In WSI, these context embeddings are clustered to find the word senses present in the text. ICE is based on a novel method for combining word embeddings using continuous Skip-gram, based on both semantic and a temporal aspects of context words. ICE is evaluated both in a new system, and in an extension to a previous system for WSI. In both cases, we surpass previous state-of-the-art, on the WSI task of SemEval-2013, which highlights the generality of ICE. Our proposed system achieves a 33% relative improvement.

## 1 Introduction

Ambiguity is pervasive in natural language and this is particularly true of word meaning: a word string may refer to several different concepts or senses. *Word sense induction* (WSI) is the problem of using a text corpus to automatically determine 1) the inventory of senses, and 2) which sense a particular occurrence of a word belongs to. This stands in contrast to the related task of *word sense disambiguation* (WSD), which is concerned with linking an occurrence of a word to an external sense inventory, e.g. WordNet. The result of a WSI system is a set of local sense labels, consistent within the system but not linked to a universal set of labels. A wide range of applications have been proposed where WSI could be useful, ranging from basic linguistic and lexicographical research (Nasiruddin et al., 2014), machine reading (Etzioni et al., 2006) and information retrieval (Véronis, 2004). WSI is of particular interest in situations where standard lexical resources are unreliable or inapplicable, such as when tracking changes of word meaning over time (Mitra et al., 2014).

According to the distributional hypothesis (Harris, 1954), word meaning is reflected in the set of contexts in which a word occurs. This intuition makes it natural to operationalize the meaning of a word – and of its contexts – using a vector-space representation, where geometric proximity corresponds to similarity of meaning. A common approach used in several successful WSI systems is to apply this geometric intuition and represent each context of a polysemous word as a vector, look for coherent clusters in the set of context vectors, and let these define the senses of the word. This approach was pioneered by Schütze (1998) using second order co-occurrences to construct the context representation. It is clear that in order to be useful in a WSI system, a geometric representation of context meaning must be designed in a way that makes clusters distinct.

Recently, neural embeddings, such as the popular Skip-gram model (Mikolov et al., 2013a), have proven efficient and accurate in the task of embedding words in vector spaces. As of yet, however, neural embeddings have not been considered for representing contexts in WSI. The systems that seem



*Proceedings of NAACL-HLT 2015*, pages 25–32,
Denver, Colorado, May 31 – June 5, 2015. 

most relevant in this context are those that train *multi-prototype* embeddings: more than one embedding per word (Huang et al., 2012). In particular, Neelakantan et al. (2014) described a modified Skip-gram algorithm that clusters instances on the fly, effectively training several vectors per word. However, whether this or any other similar approach is useful if considered as a WSI system is still an open question, since they have never been evaluated in that setting.

We make the following contributions: (1) We define the *Instance-context embedding* (ICE), a novel way for representing word instances and their context. ICE combines vectors representing context words using a novel weighting schema consisting of a semantic component, and a temporal component, see Section 3. (2) We propose two methods for using our embeddings in word sense induction, see Section 4. The first adopts a batch clustering scheme, where senses are induced after the word embeddings are computed. The number of senses is automatically chosen, based on data. The second extends an existing method for simultaneous embedding and clustering of words (Neelakantan et al., 2014). We show that our extension substantially improves the model. (3) We evaluate both proposed methods in the WSI task. We show that the two components of our proposed weighting schema both contribute to an increased overall performance. Further, we compare our method to state-of-the-art methods on Task 13 of SemEval-2013, achieving a 33% relative improvement see, Section 6.

## 2 Context clustering

Context clustering is an approach to WSI in which each instance of a word is represented by its context, embedded in a geometric space. These *context embeddings* are then clustered to form centroids representing the different senses of the target word. The context clustering approach was pioneered by Schütze (1998) who used second order co-occurrences to construct the context embedding. In this setting, the output of a WSI system is a set $\mathcal{S}_w = \{\mathbf{s}_{w,1}, \ldots, \mathbf{s}_{w,k}\}$ of $k$ locally defined senses of a word $w$, with corresponding sense embeddings $\mathbf{s}_{w,j}$. We refer to $\mathcal{S}_w$ as the *induced sense inventory* of $w$. The WSI problem is often paired with the related task of *word sense disambiguation* (WSD), concerned with linking a previously unseen occurrence of a word to an existing sense inventory. Given an instance $w_i$, of a possibly polysemous word, let its context be represented by an embedding, $\mathbf{c}_i$. The sense of $w_i$ is determined by finding the nearest neighbor to $\mathbf{c}_i$, in the sense inventory $\mathcal{S}_{w_i}$,

$$sense(w_i) = \underset{j \,:\, \mathbf{s}_j \in \mathcal{S}_{w_i}}{\arg\min}\; d(\mathbf{c}_i, \mathbf{s}_j)\;, \qquad (1)$$

where $d(\cdot, \cdot)$ is some distance function. In this work, $d$ is the cosine distance $d(\mathbf{x}, \mathbf{y}) = 1 - \mathbf{x}^T\mathbf{y}/(\|\mathbf{x}\|\|\mathbf{y}\|)$. We proceed to review *distributed word embeddings*, used in this work to create context embeddings.

### 2.1 Distributed word embeddings

A word embedding is a continuous vector representation that captures semantic and syntactic information about a word. Such representations are often based on the *distributional hypothesis* of Harris (1954), stating that the meaning of a word is largely determined by the contexts in which it appears. For word embeddings, this is realized by assigning similar embeddings to words that appear in similar contexts. These representations can be used to unveil multiple dimensions of similarity between words, such as number, topic and gender (Mikolov et al., 2013b). Word embeddings computed using neural networks were introduced by Bengio et al. (2003) and are often called *neural word embeddings*.

*Continuous Skip-gram* is an algorithm for computing word embeddings that was introduced by Mikolov et al. (2013a). This model has received a lot of attention recently, being one of the models used in the software package *word2vec* (Mikolov, 2013). The model is trained to predict the context surrounding a given *target* word. Each word $w$ is represented by two vectors, one for when the word is the target, denoted $\mathbf{u}_w$, and one for when it is in the context of another word, denoted $\mathbf{v}_w$.

We follow the interpretation of the negative sampling method for Skip-gram in Levy and Goldberg (2014). Let $D$ denote the observed data, as a set of pairs of target and context words. Then, the probability of observing the pair $(w_c, w_i)$ of a context

word $c$ and target word $i$ in the data is,

$$p((w_c, w_i) \in D) = \frac{1}{1 + e^{-\mathbf{v}_c^T \mathbf{u}_i}} , \qquad (2)$$

where $\mathbf{u}_i$ is the vector representation of the target word $w_i$ and $\mathbf{v}_c$ is the vector representation of the context word $w_c$. The vectors $\mathbf{u}_i$ and $\mathbf{v}_c$ are referred to as *word embeddings* and *context-word embeddings* respectively. Training of the Skip-gram model with negative sampling corresponds to finding embeddings that maximize $p((w_c, w_i) \in D)$ for observed context pairs and $p((w_c, w_i) \notin D)$ for random (negative) context pairs. This is usually achieved using stochastic gradient descent.

### 2.2 Clustering word instances

Clustering of vector-valued observations is a well-studied subject. Perhaps the most widely used algorithm for this purpose, $k$-means clustering, embodies many of the intuitions and difficulties of the problem. In our setting, the vectors to cluster represent instances of a single word and $k$ corresponds to the number of senses of the word. Clearly, $k$ is highly dependent on the word, and is not easily set by hand. Although many algorithms have been proposed to solve the problem for a given $k$, choosing $k$ itself remains a problem in its own right. The frequently used Gap statistic (Tibshirani et al., 2000) gives a method for solving this problem. Unfortunately, it can be prohibitively slow for use in repeated clustering of large numbers of points, as the method relies on Monte Carlo simulations. Pham et al. (2005) proposed an alternative method in which a function defined by the cluster distortion for different values of $k$, is used to evaluate cluster quality.

In the setting described above, the embeddings are assumed to be computed before clustering into senses. In contrast, Multi-sense Skip-gram (MSSG) (Neelakantan et al., 2014) attempts to learn several embeddings of a word, one for each of its different senses, by extending the Skip-gram method of Mikolov et al. (2013a). This involves a simultaneous embedding and clustering of word instances. A drawback is that their method limits the training of multi-sense embeddings to the $M$ most common words, forcing a complete re-training of the model should a new word of interest appear.

## 3 Instance-context embeddings

We propose a new method for creating context embeddings for WSI. The embeddings are based on word embeddings and context-word embeddings computed using the Skip-gram model as described in Section 2.1. Our method differs from previous approaches in that it assigns different weights to the context words based on their influence on the meaning of the target word.

More precisely, the context embedding ($\mathbf{c}$) for word instance $i$ is computed as the weighed average of the context-word embeddings representing surrounding words

$$\mathbf{c}_i = \frac{1}{Z} \sum_{\substack{-T \leq c \leq T \\ c \neq 0}} \psi_{i,c} \mathbf{v}_c \, . \qquad (3)$$

Here, $\psi_{i,c}$ is the weight for context word $c$, $\mathbf{v}_c$ is the context-word embedding for the same word and $T$ is the number of words, to the left and right, which are considered part of the context of target word $i$. $Z$ is a normalizing factor to put $\mathbf{c}_i$ on the unit sphere.

Perhaps the simplest weighting schema is the uniform, or *non-informative* schema, $\psi_{i,c}^{\text{uniform}} = \frac{1}{2T} \forall i, c$. Context embeddings using uniform weights were used in the Multi-Sense Skip-Gram (MSSG) model by Neelakantan et al. (2014) for computing sense embeddings. However, in the context of WSI it is not hard to imagine a situation where an informed weighted sum would perform better. For example, in the phrase "the rock band" the word "band" is clearly more indicative for the sense of "rock" than the word "the", and should therefore have a larger impact on the instance representation. To address this caveat we propose context embeddings based on a novel weighting schema, *Instance-context embeddings* (ICE), that leverages co-occurrence statistics naturally captured by the Skip-gram model.

### 3.1 Semantic context weights

The first component of ICE is based on the assumption that context words that strongly correlate with the target word is more important for the meaning of the target word. In the example phrase from Section 3, the word "band" is clearly a strong indicator for the presence of the word "rock", while the word

"the" occurs everywhere in English text and will therefore not have a strong correlation with "rock".

To leverage this idea, we use the Skip-gram output probability, see (2), to weight context words by

$$\psi_{i,c}^{\text{semantic}} = \frac{1}{1 + e^{-\mathbf{v}_c^T \mathbf{u}_i}}, \tag{4}$$

where $\mathbf{v}_c$ is the context-word embedding for the word $c$, and $\mathbf{u}_i$ is the word embedding of target word $i$. Using $\psi^{\text{semantic}}$ in (3) has the effect of assigning bigger importance to context words that have a semantic relation to the target word. Context words that are not useful in characterizing the sense of the target are weighted less. This is in stark contrast to the uniform weighting schema.

Levy and Goldberg (2014) discovered an interesting connection between the Skip-gram model and Pointwise Mutual Information (PMI) (Church and Hanks, 1990). Consider the optimizers of the Skip-gram objective, word and context-word embeddings, $\mathbf{u}_i$, $\mathbf{v}_c$, trained using $k$ negative samples. Levy and Goldberg showed that for sufficiently large dimensionality, these vectors satisfy the following relation, $\mathbf{u}_i{}^T \mathbf{v}_c = PMI(w_i, w_c) - \log k$ . Let $\sigma(\cdot)$ be the logistic function. For vectors satisfying the conditions stated above, we have $\psi_{i,c}^{\text{semantic}} = \sigma(PMI(w_i, w_c) - \log k)$ , establishing a connection between the semantic weights applied to Skip-gram embeddings, and PMI, a function frequently used for measuring word similarity (Pantel and Lin, 2002).

### 3.2 Temporal context weights

Window functions are used to extract local information from a sequence. In the context of NLP this translates to extracting a phrase of a given length from a larger text. The most common window function used in WSI is the rectangular window function, where $T$ words are extracted from each side of the target word. However, this approach is not optimal. In part, because it ignores the distance between the target word and the context word, but also because the sharp border makes the approach more noisy with respect to the chosen $T$.

To address these issues we instead apply a triangular window function to the context. This is inspired by the Skip-gram model, where this is achieved by uniformly sampling the context width $\in \{1 \dots T\}$. In our model we weight the context words according to target word distance as

$$\psi_{i,c}^{\text{temporal}} = \frac{1}{T} \max(0, T - |i - c|) \, . \tag{5}$$

### 3.3 Instance-context embeddings (ICE)

Finally, by combining the results of Section 3.1 and 3.2 we arrive at the definition of our proposed weighting schema

$$\psi_{i,c}^{\text{ice}} = \psi_{i,c}^{\text{semantic}} \psi_{i,c}^{\text{temporal}} \, . \tag{6}$$

## 4 Word sense induction using ICE

We devise two methods for performing word sense induction using ICE. The first is based on the $k$-means clustering algorithm. Here, word and context-word embeddings are computed using Skip-gram. Then, context embeddings are computed for all instances of a word, according to (3), and clustered using $k$-means, with Pham's heuristic for choosing $k$ (Pham et al., 2005), to form centroids representing word senses. As clustering is performed in batch, after embedding, we refer to this method as ICE-kmeans.

The second method is an extension of the MSSG model (Neelakantan et al., 2014), in which we during training of the model embed word instances using ICE. This improves the disambiguation performed at every iteration of MSSG. As this method performs the clustering in an *online* fashion, we refer to this method as ICE-online. For this, we have modified the code provided by Jeevan Shankar[1].

## 5 Evaluation

We evaluate our methods for word sense induction on shared task 13 of SemEval-2013, *Word Sense Induction for Graded and Non-Graded Senses* (Jurgens and Klapaftis, 2013). Henceforth, we let "SemEval-2013" refer to this specific task. We also investigate the influence of our weighting schema on both methods. Further, we study qualitative properties of the word instance embeddings produced by our method.

[1] https://bitbucket.org/jeevan_shankar/multi-sense-skipgram/

### 5.1 SemEval-2013, Task 13

The SemEval-2013 (test) data contains 4664 instances, each inflections of one of 50 lemmas (Jurgens and Klapaftis, 2013). The competition included both single-sense instances and instances with a graded mixture of senses. Because the manual annotations were deemed too poor, only 10% of instances were labeled with multiple senses (Jurgens and Klapaftis, 2013), which led the organizers to publish results both for all instances, and for single-sense instances only. For this reason, we consider only single-sense instances. Each instance is represented by a phrase, annotated with part-of-speech (POS) tags, comprising the word for which to determine the sense, and its context.

The rules of SemEval-2013 allowed the use of a specific corpus, ukWaC, for training of the submitted models. We have cleaned this corpus, removing formatting and making it lowercase. We extract common $n$-grams from the corpus and include them as entities in our vocabulary, e.g. Kuala Lumpur $\rightarrow$ Kuala_Lumpur. Frequency thresholds were set to 10 times for $n = 1$, 20 times for $n = 2$, and 50 times for $n \in \{3, 4\}$. Longer phrases are not considered. Following SemEval-2013, we evaluate systems for *unsupervised* WSI using two different scores, Fuzzy B-Cubed (FBC) and Fuzzy Normalized Mutual Information (FNMI) (Jurgens and Klapaftis, 2013). FBC compares two *fuzzy* covers, clusterings of the data with partial memberships, on a per-item basis. The score is sensitive to cluster size skew. FNMI is a generalization of normalized mutual information for fuzzy covers. It measures the dependence between two clusterings independently of cluster sizes. As as a final, combined score, we compute the harmonic mean (HM) of FBC and FNMI. To allow direct comparison with published results, we use the fuzzy measures even though we only consider single-sense instances.

We compare our results to two baselines from SemEval-2013. "One sense" predicts that all instances have the same sense. "One per instance" predicts that every instance has its own sense.

### 5.2 Experimental setup

For ICE-kmeans, we train a 300 dimensional Skip-gram model on the ukWaC corpus using standard parameter settings. I.e. context width set to 20 (10 before and 10 after), and 10 negative samples. We let the model iterate over the training data 9 times to improve the embeddings. For sense induction, we sample 1800 instances of very target word at random, from the ukWaC corpus. Using more instances did not improve the results in our experiments, however, for larger datasets this might not be valid. To remain general, we opted not to use the POS tags available in ukWaC, even though using them might have improved the result. Also, due to the noisy nature of the corpus, we exclude contexts where more than 30% of the words contain non-alphabetic characters. We cluster the selected instances using $k$-means clustering with the heuristic of Pham et al. (2005) for choosing $k$. For both ICE-kmeans and ICE-online, when computing the ICE vectors, the context width for was set to 20 when using the full schema, see (6), and 10 otherwise, as the full schema is less sensitive to irrelevant context words. For the MSSG part of ICE-online, we use the parameters reported in Neelakantan et al. (2014).

### 5.3 Current state-of-the-art

We compare the perfomance of our system to that of state-of-the-art systems for WSI.

First, we compare to the systems with the current best results on SemEval 2013 task 13 for single-sense word instances, AI-KU and unimelb. AI-KU (Baskaya et al., 2013) uses an approach based on substitute word vectors, inferred using a statistical language model. AI-KU achieved the highest FNMI score of the systems submitted to SemEval-2013. unimelb (Lau et al., 2013), who achieved the highest FBC score at SemEval-2013, is a system based on the topic model Latent Dirichlet Allocation and its non-parametric equivalent, Hierarchical Dirichlet Processes. Word instances are clustered based on the topic distributions inferred by the model.

The related problem of training neural embeddings of polysemous words was addressed by Huang et al. (2012) and subsequently by Neelakantan et al. (2014) with the model Multi-sense Skip-gram (MSSG), see Section 2.2. As a second experiment we extend MSSG for WSI. MSSG has not previously been used for WSI, however it produces one word embedding for each word sense, and performs a simple disambiguation procedure during training.

MSSG is thus a natural candidate for comparison. We use the standard variant of MSSG, as it achieved the best overall results in the original paper Neelakantan et al. (2014). MSSG disambiguates instances by assigning them to the sense with embedding closest to the average context-word vector of the instance, i.e. using uniform weighting. We use the parameters reported in Neelakantan et al. (2014), with the number of senses per word set to 3. MSSG takes a parameter $M$ specifying the number of words for which multiple sense vectors are computed. Like in Neelakantan et al. (2014), we set this parameter to $M = 30000$. We note that only 43 out of 50 lemmas in SemEval-2013 were in the $M$ most common words assigned multiple vectors by the MSSG methods. For the remaining 7 words, a single sense was predicted. Making sure all relevant words are included is not trivial in practice, without knowledge of the test set, as the training time of the model depends greatly upon $M$.

## 6 Results

We report the results of all experiments below.

### 6.1 Qualitative evaluation of instance vectors

Consider the word "paper". WordNet (Miller, 1995) lists seven senses of "paper" as a noun: 1) a medium for written communication, 2) an essay (especially one written as an assignment), 3) a scholarly article describing the results of observations or stating hypotheses, 4) a daily or weekly publication on folded sheets; contains news and articles and advertisements, 5) a business firm that publishes newspapers, 6) a material made of cellulose pulp derived mainly from wood or rags or certain grasses, 7) the physical object that is the product of a newspaper publisher. Assigning an instance to one of these senses can be challenging even for a human reader.

The word "paper" is one of the 50 lemmas in the SemEval-2013 evaluation data with corresponding instances that cover six of the senses listed in WordNet. In Figure 1, we show context embeddings for these instances, plotted using the dimensionality reduction tool t-SNE (Van der Maaten and Hinton, 2008). Figure 1a represents context embeddings computed using a uniform average, and Figure 1b plots the instance context embeddings computed with using ICE, as described in Section 3. The colors and markers correspond to gold-standard WordNet annotations provided by SemEval. The size of a marker in Figure 1b is proportional to the average ICE weight of words in the context of an instance and is indicative of the confidence in the instance vector. A low average ICE weight indicates that the context is not predictive of the target word.

For the senses, "material", "scholarly article", "newspaper" and "essay", the instances in Figure 1b are noticeably more clustered than in Figure 1a. This shows that the senses of these words are better represented using ICE weighting for context embeddings than a uniform schema.

### 6.2 Semeval WSI results

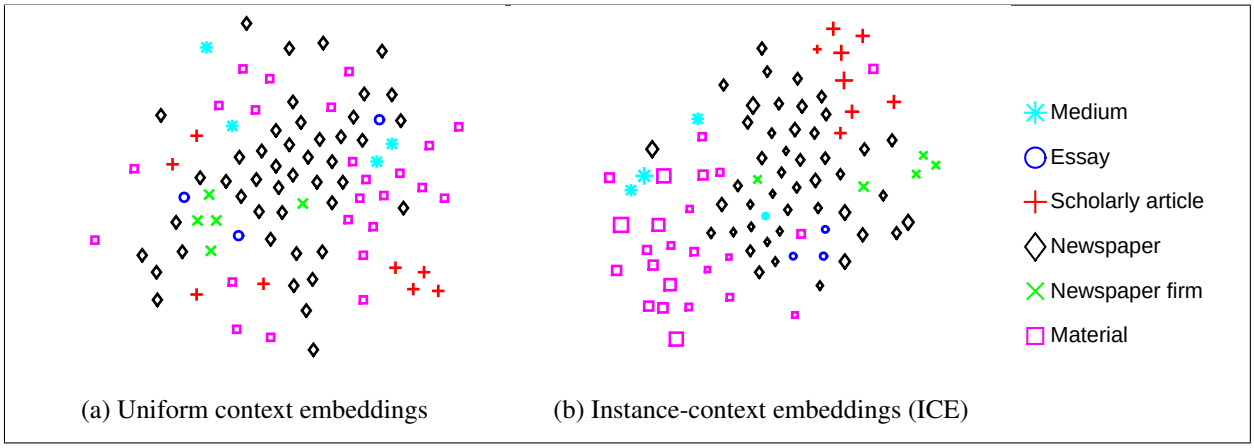
The results of the WSI evaluation on shared task 13 of SemEval-2013 are presented in Table 1. Here, our system ICE-kmeans, and our MSSG extension ICE-online, use the ICE weighting schema, see (6). MSSG is the system presented in Neelakantan et al. (2014) without modifications. AI-KU and unimelb represent the best systems submitted to SemEval-2013, and AI-KU the current state-of-the-art in WSI.

First, we note that ICE-kmeans achieves the overall best results with respect to both scores, corresponding to a relative improvement of 31.1% in FNMI and 15.9% in FBC. Further we note that the previous best FBC and FNMI belong to different methods. This is important since, as with precision and recall, achieving a high score in one of these measures can be achieved using a trivial baseline, see the first two methods in Table 1. Hence, a better benchmark, analogue to the $F_1$-score, is the harmonic mean (HM) of the two complementary scores. Considering this our results are even more impressive with a 33% relative improvement.

### 6.3 Semantic and temporal component of ICE

We evaluate the impact of using context embeddings based on the different weighting schemas defined in Section 3, over embeddings based on uniform weights. The results are presented, as harmonic mean and relative improvement over previous state-of-the-art AI-KU, in Table 2.

First, we note that both variants of our full system (ICE) offers a substantial relative improvement over AI-KU. We note that the results are always




(a) Uniform context embeddings (b) Instance-context embeddings (ICE)

Figure 1: Context embeddings for instances of the noun "paper" in the SemEval-2013 test data, plotted using t-SNE. The legend refers to WordNet gold standard embeddings.

| Method | FBC(%) | FNMI(%) | HM |
|---|---|---|---|
| One sense | 57.0 | 0 | 0 |
| One per instance | 0 | 4.8 | 0 |
| Unimelb | 44.1 | 3.9 | 7.2 |
| AI-KU | 35.1 | 4.5 | 8.0 |
| MSSG | 45.9 | 3.7 | 6.8 |
| ICE-online | 48.7 | 5.5 | 9.9 |
| ICE-kmeans | **51.1** | **5.9** | **10.6** |

Table 1: Results for single-sense instances on the WSI task of SemEval-2013. HM is the harmonic mean of FBC and FNMI.

better when using semantic weights, Eq (5), over uniform, and always *best* when using the full ICE schema, Eq (6). These results clearly indicate that both the semantic and temporal weight components contribute to a better system. We note that the $k$-means systemis consistently better than the online version. This conforms to expectations as the online system has access to less information at every cluster assignment. The two top left results (in gray) correspond to the original MSSG system.

## 7 Conclusion

We have presented Instance-context embedding (ICE), a method for embedding word instances and their context for use in word sense induction (WSI). At the heart of the system are instance representations based on neural embeddings of context words, combined using a novel weighting schema.

We have shown that ICE is sucessful in representating instances of polysemous words, not just in our own WSI system, but in an extension of an existing model as well. In an evaluation on the WSI task of SemEval-2013, our system beat previous state-of-the-art methods, achieving a 33% relative improvement. Further, we have established the benefits of using ICE over a uniform weighting schema, by showing empirically that each of its components contribute to a more accurate system.

## Acknowledgments

The authors would like to acknowledge the project *Towards a knowledge-based culturomics* supported by a framework grant from the Swedish Research Council (2012–2016; dnr 2012-5738), and the project *Data-driven secure business intelligence* grant IIS11-0089 from the Swedish Foundation for Strategic Research (SSF). The third author was supported by Swedish Research Council grant 2013–4944, *Distributional methods to represent the meaning of frames and constructions*.

## References

Osman Baskaya, Enis Sert, Volkan Cirik, and Deniz Yuret. 2013. Ai-ku: Using substitute vectors and co-occurrence modeling for word sense induction and disambiguation. *Atlanta, Georgia, USA*, page 300.

| | Uniform | | ICE (Sem. only) | | ICE | |
|---|---|---|---|---|---|---|
| **Method** | **HM** | **Impr.(%)** | **HM** | **Impr.(%)** | **HM** | **Impr.(%)** |
| ICE-online | 6.8† | -14† | 7.9 | -1.1 | 9.9 | 24 |
| ICE-kmeans | 7.6 | -5.2 | 9.6 | 20 | **10.6** | **33** |

Table 2: Impact of the two different components of ICE on SemEval-2013 task 13 (single-sense). Imprv. is the relative improvement over the state-of-the-art system AI-KU (with an HM of 8.0). †Gray numbers correspond to the original MSSG system. The weighting schemas are: **Uniform**, Section 3, **ICE (Semantic only)**, Eq (4), and **ICE**, Eq (6).


Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29, March.

Oren Etzioni, Michele Banko, and Michael J Cafarella. 2006. Machine reading, proceedings of the 21st national conference on artificial intelligence.

Zellig Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

Eric Huang, Richard Socher, Christopher Manning, and Andrew Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 873–882, Jeju Island, Korea, July. Association for Computational Linguistics.

David Jurgens and Ioannis Klapaftis. 2013. Semeval-2013 task 13: Word sense induction for graded and non-graded senses. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 2, pages 290–299.

Jey Han Lau, Paul Cook, and Timothy Baldwin. 2013. unimelb: Topic modelling-based word sense induction. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 2, pages 307–311.

Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2177–2185. Curran Associates, Inc.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Workshop at ICLR*.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *Proceedings of NAACL-HLT*, pages 746–751.

Tomas Mikolov. 2013. word2vec. `https://code.google.com/p/word2vec/`.

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

Sunny Mitra, Ritwik Mitra, Martin Riedl, Chris Biemann, Animesh Mukherjee, and Pawan Goyal. 2014. That's sick dude!: Automatic identification of word sense change across different timescales. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1020–1029, Baltimore, Maryland, June. Association for Computational Linguistics.

Mohammad Nasiruddin, Didier Schwab, Andon Tchechmedjiev, Gilles Sérasset, Hervé Blanchon, et al. 2014. Induction de sens pour enrichir des ressources lexicales. In *Actes des 21ème conférence sur le Traitement Automatique des Langues Naturelles (TALN 2014)*.

Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proceedings of EMNLP*.

Patrick Pantel and Dekang Lin. 2002. Discovering word senses from text. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 613–619. ACM.

Duc Truong Pham, Stefan S Dimov, and CD Nguyen. 2005. Selection of k in k-means clustering. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 219(1):103–119.

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational linguistics*, 24(1):97–123.

Robert Tibshirani, Guenther Walther, and Trevor Hastie. 2000. Estimating the number of clusters in a dataset via the gap statistic. 63:411–423.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85.

Jean Véronis. 2004. Hyperlex: lexical cartography for information retrieval. *Computer Speech & Language*, 18(3):223–252.