



**HAL**  
open science

# Word Sense Induction with Attentive Context Clustering

Moshe Stekel, Amos Azaria, Shai Gordin

► **To cite this version:**

Moshe Stekel, Amos Azaria, Shai Gordin. Word Sense Induction with Attentive Context Clustering. *Journal of Data Mining and Digital Humanities*, 2022, NLP4DH, 10.46298/jdmdh.9175. hal-03586559v4

**HAL Id: hal-03586559**

**<https://hal.science/hal-03586559v4>**

Submitted on 6 Jun 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Word Sense Induction with Attentive Context Clustering

Moshe Stekel<sup>1</sup>, Amos Azaria<sup>1</sup>, Shai Gordin<sup>2</sup>

<sup>1</sup>Computer Science Dpt., Ariel University, Israel

<sup>2</sup>Digital Pasts Lab, Land of Israel Studies and Archaeology Dpt., Ariel University, Israel

Corresponding author: Moshe Stekel, [mstekel@gmail.com](mailto:mstekel@gmail.com)

## Abstract

This paper presents ACCWSI (Attentive Context Clustering WSI), a method for Word Sense Induction, suitable for languages with limited resources. Pretrained on a small corpus and given an ambiguous word (a query word) and a set of excerpts that contain it, ACCWSI uses an attention mechanism for generating context-aware embeddings, distinguishing between the different senses assigned to the query word. These embeddings are then clustered to provide groups of main common uses of the query word. We show that ACCWSI performs well on the SemEval-2 2010 WSI task. ACCWSI also demonstrates practical applicability for shedding light on the meanings of ambiguous words in ancient languages, such as Classical Hebrew and Akkadian. In the near future, we intend to turn ACCWSI into a practical tool for linguists and historians.

## Keywords

word sense induction; wsi; word sense discrimination; wsd; natural language processing; nlp; clustering; semitic languages; hebrew; akkadian;

## I INTRODUCTION

Natural language expresses human concepts, thoughts, emotions, and insights. That is, natural language represents a model of extremely high complexity - the human mind (at least, its communication-driven layers). Some researchers believe that natural language is an environment in which compromise is inevitable when projecting the infinite number of dimensions of human thinking onto the much smaller number of dimensions of human speech Fedorenko and Varley [2016]. The multiplicity of the meaning of a single word, such as polysemy (similarity obtained from a common source) or homonymy (accidental similarity), is, therefore, an expected product of this compromise. Below are two common examples of word sense ambiguity:

- “I can hear *bass* sounds” versus “They like grilled *bass*”
- “We crossed the river to the other *bank*” versus “Mike deposited the money in his *bank* account”

Humans are able to disambiguate the polysemy/homonymy or understand contextual nuances by using the clues that come from the context of the ambiguous word. One of the fundamental tasks of natural language processing is Word Sense Induction (WSI), a task of automatic discrimination of the different senses of words by finding these contextual clues.

It is difficult to overestimate the importance of accurate Word Sense Induction when dealing with common Natural Language Processing (NLP) tasks, such as Information Retrieval or Search Clustering. Furthermore, historical research seeks to correctly induce the meaning of

words in order to resolve doubts about many historical issues. As a good example, we can refer to the Akkadian lemma “galû”, the meaning of which ranges between the negative shade of “exile” or “deportation”, the neutral shade of “relocation” and the rare positive one of “appointment”. Another example is the Hebrew lemma “zakar”, which takes on both the meanings of “memory” and “male”. Accurate Word Sense Induction is essential for the correct understanding of ancient documents.

In this paper, we present an Attentive Context Clustering WSI (ACCWSI). ACCWSI first creates a word embedding for each word, which is identical for any context that it appears in. ACCWSI uses the cosine similarity between the words in the context and the word in focus to determine the attention that each word should achieve to form a context-aware vector representation for each appearance of the word in focus. ACCWSI then clusters the resulting vectors, such that each cluster represents a different meaning of the word. ACCWSI has demonstrated high practical applicability in languages with limited resources and obtained a very high score by the evaluation framework of SemEval-2 2010 Task 14 Manandhar et al. [2010]. ACCWSI achieved a high score not only with the original training dataset but also with a training dataset reduced to a fraction of 2.6% of the original dataset, which is comparable to the size of the Hebrew Bible.

## II RELATED WORK

Word Sense Induction and Word Sense Disambiguation provide fertile ground for researchers, starting from very early attempts to tackle these non-trivial tasks, such as “simulated annealing” according to human-edited dictionary Cowie et al. [1992] and employing the “conceptual distance” between contexts Agirre and Rigau [1996], going through later unsupervised methods, that use patterns of word co-occurrence Bordag [2006] or bigrams of web search results Udani et al. [2005], continuing with “hidden concepts” of the contextual words, that not necessarily overlap with the sense of the ambiguous word Chang et al. [2014], and ending with the most recent solutions like Eyal et al. [2021], that uses word substitutions of modern Masked Language Models, such as Google BERT MLM. In addition, the technique of pipelining a clustering algorithm and cosine distance (note that we do this in reverse order) is used by Hämäläinen and Alnajjar [2019] for dividing words (not necessarily polysemous or even similar) into semantic fields. This idea is not directly related to the Word Sense Induction task but it is another approach to dividing a given text into different semantic groups by using clustering and cosine distance. Our research was inspired by two main works: the context-group discrimination algorithm Schütze [1998] from the Context Clustering category and the Google BERT language model Vaswani et al. [2017]. Amrami and Goldberg Amrami and Goldberg [2019] utilize Google BERT for their WSI method. However, their method does not meet our requirement of being able to induce word senses in languages with limited resources, as training Google BERT on small corpora does not provide sufficient accuracy Ezen-Can [2020]. The high scores achieved by the BertWSI model in the SemEval-2 2010 Task 14 Manandhar et al. [2010] metrics are credited to the fact that the underlying model was pre-trained by Google on a huge corpus of text. Our solution takes advantage of the basic mechanism of attention Galassi et al. [2020] underlying BERT without applying the complex process of learning attention weights and thus achieves good results when applied to small datasets. The only weight learning process we use is the Word2Vec Goldberg and Levy [2014] model training that requires far fewer resources than attention-based learning. Thus, we provide a practical tool for the study of the meanings of words in resource-limited languages, such as ancient dead languages. The Clustering by Committee work Pantel and Lin [2002] gave us the idea to use a threshold of 0.5 as an acceptable proportion of orphan instances when measuring the quality of a clustering solution (see Section

3.4.3). We also explored Lin’s algorithm Lin [1998], which uses the word clustering approach by combining words with similar semantics into sense representations, but it was found less effective when it came to discriminating senses of words in resource-constrained languages.

### III TASK AND ALGORITHM

#### 3.1 WSI task definition

The general definition of WSI is automatic detection of the set of senses denoted by a word. A simplified version of WSI can be defined as follows: given a list of lemmatized sentences and a query lemma, find all the sentences in the list that contain the query lemma, and group them so that the instances of the query lemma in one group are semantically similar to each other and noticeably different from the instances in other groups. This is a simplified definition because, when lemmatizing, we ignore some input information, such as the part of speech, tense, etc. Note that ignoring the part of speech information of the target word is attractive, especially for ancient genres in which the archaic syntactic forms of words may provide no part of speech information (for instance refer to some hardly explainable verses of the Hebrew Psalms).

#### 3.2 Attention mechanism

Our method uses the following “basic attention” mechanism: given a target word (query) and its “context”, either the whole sentence or some “window” of words containing the query word, each element of the context is evaluated by its cosine similarity to the query word. The result is optionally multiplied by a constant factor and eventually softmaxed. We refer to the result as the “weights of similarity” or “weights of relevance”. The semantically closer two words are, the greater is the cosine similarity between their embeddings and, therefore, the appropriate weights of relevance are greater. The original word embeddings of the context members are multiplied by the appropriate weights of relevance and thus the power of every context member is improved or worsened according to its relevance to the query word. When these new context-sensitive embeddings are summed into a single vector, this sum represents a context-aware vector of the query word that embeds its “local sense” with respect to this specific context, where the relevance of each context member is taken into consideration. Figure 1 illustrates this mechanism.

#### 3.3 The ACCWSI algorithm

We now present our Attentive Context Clustering WSI (ACCWSI) algorithm. The ACCWSI algorithm (see Algorithm III.1) first replaces the lemmas with their Word2Vec embeddings Goldberg and Levy [2014]. It then uses the attention mechanism described above (Section 3.2), resulting in context-aware vectors, that are used as input to the DBSCAN clustering algorithm Schubert et al. [2017], producing clusters of different “shades of meaning” of the query lemma. Since different contexts are best defined by different most relevant context members, and conversely - similar contexts are defined by similar context members, the result vectors can be easily clustered. Figure 2 illustrates this idea.

Let’s explain the algorithm in detail: line 2 runs the language model creation - the process that generates multi-dimensional vectors that correspond to the words in the text. The distances between the vectors are supposed to express the semantic distances between the corresponding words. Line 3 finds the sentences in which the query lemma occurs. Lines 5-11 iterate over these sentences, compute the similarity of their members to the query lemma (line 8), normalize the similarity values (softmax - line 9), and then use the normalized values to aggregate the weighted values by summing them, resulting in a list of weighted context vectors. These context vectors are eventually clustered in line 12.

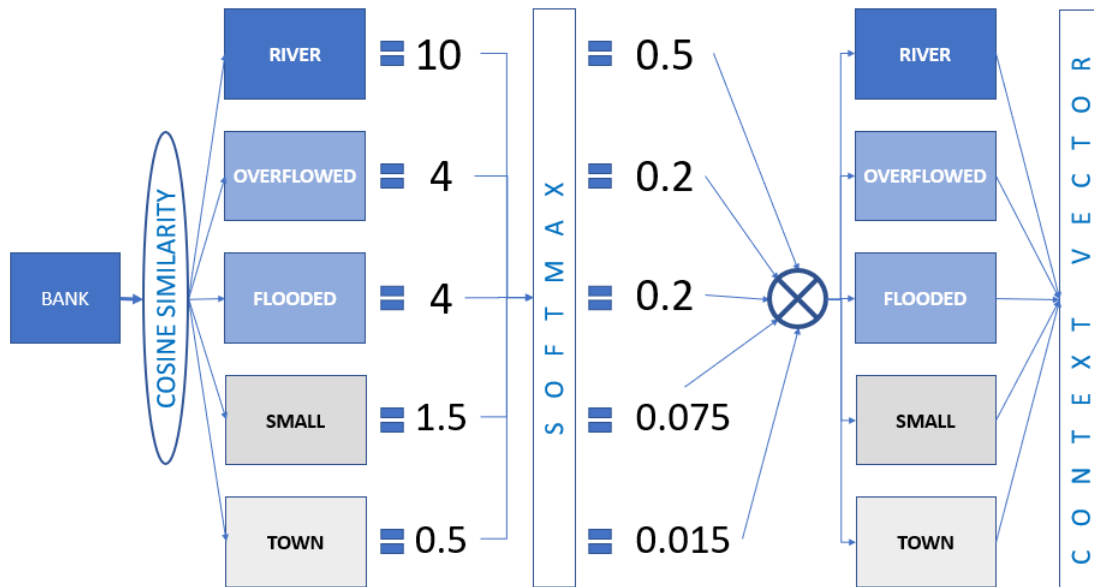


Figure 1: Illustration of the attention mechanism

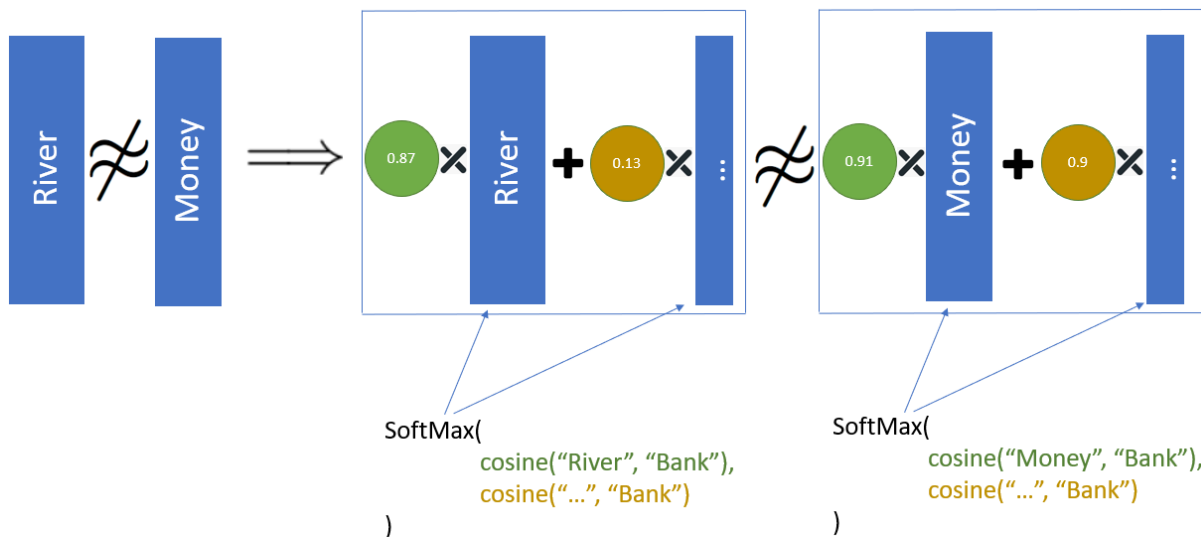


Figure 2: An illustration of separability of context-aware vectors generated by ACCWSI: the most relevant terms (green weights) with respect to the query term “bank” are “river” in the first context and “money” in the second context. They are different and therefore the result context-aware vectors are different. Less relevant terms are multiplied by smaller weights (light brown) and thus have a smaller effect on the final context-aware vector.

### 3.4 Hyperparameters

Algorithm III.1 uses several hyperparameters: Word2Vec window, the choice of the clustering algorithm, and the internal hyperparameters of the latter. The optimal values of these parameters can be found either empirically or by using well-known optimization methods. In this section, we explain these hyperparameters, briefly overview the optimization methods, and present the method that achieved the best accuracy in our case. Note that in our online tool we will use our recommendations for these parameters as their default values allowing the end-users to modify

---

**Listing III.1** ACCWSI Algorithm.

---

```
1     ACCWSI(text,lemma)
2         model := word2vec(text)
3         sentences := filter_by_lemma ( text ,lemma)
4         ctx_aware_vecs := []
5         for each s in sentences
6             ctx_vecs := model.get_vectors (s)
7             lemma_vec := model.get_single_vector (lemma)
8             sim := cosine_sim ( ctx_vecs ,lemma_vec)
9             sim_weights := softmax(sim)
10            new_lemma_vec :=  $\sum_i ctx\_vecs_i * sim\_weights_i$ 
11            ctx_aware_vecs .push(new_lemma_vec)
12        return DBSCAN(...).fit ( ctx_aware_vecs )
```

---

these default values according to the characteristics of their datasets.

### 3.4.1 *Word2Vec Window*

This parameter determines the size of the context to be scanned from each direction around the target word when training the Word2Vec model to perform the missing word prediction task (CBoW architecture) or the context prediction task (Skip-Gram architecture). The optimal value of this parameter intuitively depends on the native average “density of context” inherent to the target language. We found the optimal value empirically by iterating over the range from 2 to 10 and evaluating the result by manually checking the semantic similarity of words suggested by the model. The best values were 5 for English and 2 for Classical Hebrew. This difference is probably due to the specific syntactic structures of Classical Hebrew verses, which are statistically much shorter than the syntactic structures of typical Modern English sentences.

### 3.4.2 *The choice of the clustering algorithm*

We evaluated several clustering algorithms on our task, including KMeans Hamerly and Elkan [2004], Gaussian-Mixture model Reynolds [2009] and DBSCAN Schubert et al. [2017]. DBSCAN, the density-based clustering algorithm, performed slightly better and was therefore selected as our clustering algorithm for this paper. In our online tool, we will present a list of clustering algorithms to choose from and the end-users will be able to choose the best clustering algorithm that suits their datasets.

### 3.4.3 *DBSCAN-eps*

This parameter is a key one for the density-based clustering proposed by DBSCAN. It defines the maximum distance between two points to be considered as neighbors. There are several methods in the literature for optimizing the value of this parameter, such as the Kneedle algorithm for finding the maximum curvature in the graph of distances, the Silhouette Score for evaluating the clustering quality, and more. Although these optimization methods demonstrated good performance (unsupervised V-Measure of 15.3%), we propose a heuristic that performed better. The rationale behind the heuristic is that text can contain instances of ambiguous words with highly clear context, in addition to other instances with more obscure context. Decreasing the value of *eps* results in clearer but tighter clusters, filtering out distant “noisy” instances. In our case, narrowing the clusters while keeping the number of the “noisy” instances below 50% gave good results. Algorithm III.2 demonstrates this heuristic.



For other datasets, the heuristic above may be less effective because there should not be a direct correlation between the amount of noise and the distribution of word senses, so it can always be a good idea to let the user adjust the value of this parameter empirically. In our future online tool for researchers, summarizing this research (see Section VI), we plan to add dedicated controls to tune all ACCWSI hyperparameters.

---

**Listing III.2** Fine-tuning the DBSCAN *eps* hyperparameter - the value of *eps* is iteratively decreased until the noise (the fraction of the orphan instances) becomes greater than  $\frac{1}{2}$

---

```

1   BEST_DBSCAN_EPS(cxt_aware_vectors)
2   best_eps := 0.95
3   for each  $x \in \text{range}(90, 0, -5)$  do
4     eps :=  $x/100$ 
5     labels := DBSCAN(eps = eps).fit(cxt_aware_vectors)
6     noise := labels.count(-1)/len(labels)
7     if noise ≤ 0.5 then
8       best_eps := eps
9     else
10      break
11    end if
12  end for
13  return best_eps

```

---

## IV EXPERIMENTAL EVALUATION

We ran an experiment to evaluate the algorithm on **Sem-Eval 2010 Task 14** Manandhar et al. [2010], which aims to objectively measure and compare the quality of WSI systems. Both training and test data are English sentences containing polysemous or homonymous nouns and verbs. The goal of the task is to split the instances of each ambiguous word and their contexts into clusters representing different meanings. The result is assessed by comparison with the “Gold Standard” clustering performed by human experts. In Section 4.1 we present the Unsupervised V-Measure and F-Score metrics of this assessment as well as the Supervised Recall metric.

### 4.1 SemEval-2 2010 Task 14 Evaluation

In Task 14 of the SemEval-2 2010 workshop Manandhar et al. [2010], participants were asked to train their models on the corpus of training data provided by the organizers, and then perform word sense induction for a set of sentences containing both ambiguous nouns and ambiguous verbs. The results were assessed against the “Gold Standard” clusters compiled by human experts. The tables below show the metrics achieved with ACCWSI trained on the full training corpus (by training ACCWSI we mean training its internal Word2Vec model), as well as the metrics achieved with the reduced ACCWSI, which was trained on a randomly selected 2.6% of the training data, along with those of the participants with the highest scores in every metric.

## V APPLICATION EXAMPLES

In this section, we present examples of applying our method to a relatively small Hebrew corpus—the Hebrew Bible. We used the text-fabric version of the BHSA project to generate the appropriate dataset and run the ACCWSI algorithm on it. Figure 3 shows the operation of the ACCWSI algorithm used to obtain two different meanings of “bank” in English. Figure 4 and Figure 5 present the induced classes for two ambiguous Hebrew Biblical lemmas: **khalal**

System	VM % (All)	VM % (Nouns)	VM % (Verbs)
ACCWSI full	17.3	20.7	12.3
Hermit	16.2	16.7	15.6
UoY	15.7	20.6	8.5
KSU KDD	15.7	18	12.4
ACCWSI reduced	15.4	18.8	10.4
Duluth-WSI	9	11.4	5.7
...			
...			
Duluth-WSI-SVD-Gap	0	0	0.1

Table 1: V-Measure (VM) unsupervised evaluation. V-Measure assesses the quality of a clustering solution by explicitly measuring its homogeneity and its completeness. Homogeneity refers to the degree how much each cluster consists of data points primarily belonging to a single Gold Standard class, while completeness refers to the degree how much each Gold Standard class consists of data points primarily assigned to a single cluster. V-Measure is the harmonic mean of homogeneity and completeness.

(dead body/desecrate) and **zakar** (male/memory). The instances of the first lemma were split into 2 sense clusters while the instances of the second lemma were split into 5 sense clusters. ACCWSI seems to perform well and provide satisfactory clusters despite the small training corpus.

### 5.1 ACCWSI performance on ancient Akkadian texts

Another experiment we did was Word Sense Induction in ancient Akkadian texts. Between the 9th to the late 7th centuries BCE, the Assyrian Empire deported millions of people across the Near East. By even the most humble estimates, around 1.3 million people were moved around as a result of conquest, labour recruitment, or as punishment, just to name the central reasons for this dire process (Sano 2020). However, the records for these deportations are numerous and came down to us in different genres that deal with the act of deportation, or forced migration, from different points of view: contemporaneous Assyrian royal inscriptions, letters, and administrative texts, as well as Babylonian historical chronicles, written many years after the events in question. All were written in Assyrian and Babylonian, two close dialects of Akkadian, the oldest known (East-)Semitic language in the world. In total, 19 different verbs deal with various stages of the forced migration, like the capture of people or forced recruitment, their change of location, and resettlement. Even then, there are differences across meanings for specific verbs, sometimes minute ones, but also quite substantial in terms of semantics.

A good example of such a complicated verb is *galû* which the Chicago Assyrian Dictionary (CAD), the most comprehensive dictionary of Akkadian, translates as “1. to go into exile, 2. to deport, to exile (Š-stem, causative)” (CAD Š/3, 201). Its usage is limited to a Babylonian context, either in Assyrian letters dealing with Babylonia or Babylonian chronicles (Sano 2020, 34). As text 1 below shows, *galû*, much like Biblical Hebrew *GLY/H*, is used in consequence of a military conflict. However, a single instance in a letter from the time of Tiglath-pileser III (c. 731-730 BCE), here text 2, shows a rare nuance of the verb. Under certain political circumstances, people could ask for someone to deport them to Assyria, likely referring to the safety of being a protected refugee under the direct responsibility of the Assyrian king. This might also be the meaning of certain cases in Aramaic, where *gly* in G-stem active participle



System	FS % (All)	FS % (Nouns)	FS % (Verbs)
Duluth-WSI-SVD-Gap	63.3	57	72.4
KCDC-PT	61.8	57	72.4
...			
...			
ACCWSI reduced	55.9	51.3	62.7
...			
...			
ACCWSI full	53.8	47.2	63.4
Duluth-WSI-SVD	41.1	37.1	48.2
Duluth-WSI	41.1	37.1	48.2
...			
...			
Duluth-R-110	16.1	15.8	16.4

Table 2: Paired F-Score (FS) unsupervised evaluation: two sets of instance pairs are generated - a set of all possible instance pairs within each induced cluster and a set of all possible instance pairs within each Gold Standard class. Precision is the number of common instance pairs between the two sets to the total number of pairs in the induced clusters, while recall is the number of common instance pairs between the two sets to the total number of pairs in the Gold Standard classes. F-Score is the harmonic mean between precision and recall.

means “exile, refugee”, or in D-stem means “to emigrate”, ([Comprehensive Aramaic Lexicon, s.v. gly D and C](#)).

**Text 1:** SAA 19, 27 rev. 4’-8a’ ([online edition, Luukko 2012](#))

4’ LUGAL? lu? ú-di NIM.MA.KI-a-[a] 5’ LÚ.ERIM-MEŠ-šú-nu TA DUMU <sup>m</sup>GIN—NUMUN  
la? 6’ i-du-ku ù ša—da-a-ni 7’ ú-sag-li-šú-nu šú-nu-ú-ma 8’ ig-da-al-ú

(rev. 4’-5’) The Elamites killed their soldiers with the son of Mukin-zeri and (6’-8’) **deported them by force**. They too **went into exile**.

**Text 2:** SAA 19, 87 obv. 8b’-13a’ ([online edition, Luukko 2012](#))

8’ ... e-gir-tum ša ina UGU 9’ [<sup>md</sup>]AMAR.UTU—A—SUM-na na-u-ni-ni it-tab-lu-ni 10’ [ina]  
pa-ni-ni i-si-si-ú : ù <sup>m</sup>ba-la-su 11’ [ip]-ta-la-a a—da-niš ma-a an-nu-rig x+[x x] 12’ [at]-tu-nu  
tal-la-ka ma-a ša-ga-la-ni [o] 13’ [i]-si-ku-nu la-al-lik ...

(obv. 8’-9’) They intercepted the letter which was brought to Merodach-baladan (10’) and read it [in] our [pr]esence. But Balassu (11’) [g]ot very scared, saying: (12’) “You (pl.) must come this moment and **deport me!** (13’) I will go [wit]h you (pl.)”

Figure 7 (shortened) shows that ACCWSI produces a single cluster only for a limited number of instances of *galû* found in the [State Archives of Assyria Online at ORACC](#). In stark comparison, Figure 6 demonstrates that applying ACCWSI to the polysemous lemma *kayyānu*, which is more common in Akkadian texts, produces much better distinction of different clusters of meaning. Three possible reasons might lead to the difference in the resulting clusters between *galû* and *kayyānu*:

1. The meanings of a cultural lemma like *galû* are close to each other, at least in terms of their contexts, and even humans do not necessarily agree on them.

System	SR % (All)	SR % (Nouns)	SR % (Verbs)
ACCWSI full	63.7	59.6	71.1
ACCWSI reduced	62.7	57.5	69.8
UoY	62.4	59.4	66.8
Duluth-WSI	60.5	54.7	68.9
...			
...			
Duluth-Mix-Uni-Gap	18.7	1.6	43.8

Table 3: Supervised recall (SR) using a test set split with 80% mapping and 20% 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 one is used to evaluate methods in a WSD setting.

2. The source cuneiform tablets contain many discontinuations, breaks and names of peoples and places, providing contexts of very different quality from one another.
3. ACCWSI focuses on careful reading and does not rely on the experience gained from reading a large number of texts. Although this was previously considered its main advantage, it becomes a stumbling block when disambiguating very minute nuances.

## VI FUTURE WORK

In our future work, we will address these points without giving up the possibility of training the model on limited resources, while maintaining the main advantage of our method, which is to perform the Word Sense Induction task on a limited amount of text.

Iterating the process of generating context embeddings may improve the accuracy of the clustering. In our future work, we plan to develop a method for determining the “center of mass” (or “centroid” for convex clusters) of every cluster. These centers will be treated as new “query” embeddings and the ACCWSI attention-weighted technique will be reapplied within each cluster using its new query (its center). This should provide finer discrimination of meanings. This iterative process can be repeated many times until maximum accuracy is achieved.

The main practical goal of this study is to create an online tool for linguists and historians. This tool will allow the researcher to select a text corpus to scan, a word embedding generation algorithm to use (Word2vec/GloVe/BERT), and specify the values of various hyperparameters that control the ACCWSI flow.

## VII CONCLUSION

In this paper, we propose ACCWSI, an algorithm to automatically induce various senses of ambiguous words by automatically focusing on the most relevant words from their contexts. After learning generic word embeddings into a Word2Vec model, ACCWSI uses the basic attention technique for determining the most relevant context members and generating context-aware embeddings, each with a semantic direction that aggregates the directions of its context members. Distant meanings imply distant context embeddings and vice versa, and thus standard clustering techniques can be easily applied for grouping the context embeddings by their common semantic directions. ACCWSI has shown excellent performance even when trained on a small subset of the training data in the SemEval-2 2010 task 14. Furthermore, ACCWSI demonstrated high applicability in the disambiguation of word senses in ancient Semitic languages, such as Classical Hebrew and Akkadian.

Sentence	Attention Highlights	Cluster
Her bank account was rarely over two hundred.	account, rarely	0
After breakfast, she closed her account at the bank and turned in her resignation.	account, close, turn	0
How could a man with four million in the bank be in financial danger?	financial, man, danger	0
Seating herself on a low bank, she studied the souls.	seat, study, low	0
If you would know the history of these homesteads, inquire at the bank where they are mortgaged.	mortgage, homestead, history	0
I guess he had some bucks at one time – back when he bought all this land – but his bank account never held a candle to mine.	account, hold, buy	0
A stream bank is the terrain alongside the bed of a stream	stream, stream, bed	1
He walked up and down the river, leading his house behind him; but he kept his eyes turned always toward the dim, dark spot which he knew was the old North Church.	river, church, spot	1
She waded to the bank and picked up he shoes and stockings.	stocking, shoe, wade	1
The town of Barwani is situated near the left bank of the Nerbudda	town, near, left	1
Cushing himself swam to the swamps on the river bank, and after wading among them for hours reached a Federal picket boat.	river, boat, swamp	1
Within an hour, there were riding side-by-side down the south bank of the creek, searching for the blocked area.	creek, area, south	1

Figure 3: Two different meanings of **bank**, the financial institute and the geographic terrain, are represented by the clusters in the figure. The “attention highlight” column shows the most relevant context words. The first cluster contains an interesting failure: the fourth sentence is clustered as a financial institute even though a human would cluster it as a geographic terrain. The reason is that the most relevant context words “seat, study, low” are not sufficiently indicative.

The code of the Jupyter notebooks and other utilities we used during our research can be found in the GitHub repository below. The code resources are self-contained and reusable and can be useful in a variety of contexts<sup>1</sup>.

<sup>1</sup><https://github.com/mstekel/accwsi>

Sentence	Attention Highlights	Cluster
וּמְזַרְעֵךְ לֹא־תִתֵּן לְהַעֲבִיר לְמַלְךְ וְלֹא תַחַלֵּל אֶת־שֵׁם אֱלֹהֶיךָ אֲנִי יְהוָה.	נתן, י-ה-ו-ה, עבר	0
לָכֵן אָמַר לְבֵית יִשְׂרָאֵל כֹּה אָמַר אֲדֹנָי יְהוִה לֹא לְמַעַנְכֶם אֲנִי עֹשֶׂה בַּיִת יִשְׂרָאֵל כִּי אִם לְשֵׁם קֹדֶשׁ אֲשֶׁר חָלַלְתֶּם בְּגוֹיִם אֲשֶׁר בְּאַתֶּם שֵׁם.	גוי, ישראל, קדש	0
וְאֶחָמֵל עַל שֵׁם קֹדֶשׁ אֲשֶׁר חָלְלוּהוּ בַּיִת יִשְׂרָאֵל בְּגוֹיִם אֲשֶׁר בָּאוּ שָׁמָּה.	חמל, גוי, ישראל	0
וַיָּבֹאוּ אֵל הַגּוֹיִם אֲשֶׁר בָּאוּ שֵׁם וַיַּחַלְלוּ אֶת שֵׁם קֹדֶשׁ בְּאֶמְרָה לְהֵם עַם יְהוִה אֱלֹהֵי וּמֵאַרְצוֹ יֵצְאוּ.	עם, גוי, קדש	0
וַיִּפְּלוּ סַלְלִים בְּאַרְץ כַּשְׂדִּים וַיִּמְדְּקוּרִים בְּחוּצוֹתֶיהָ.	דקר, חוץ, נפל	1
כִּי נִתְּתִי אֶת חֲתִיתִי בְּאַרְץ סִיִּים וְהִשְׁכַּב בְּתוֹךְ עַרְלִים אֶת סַלְלֵי חֶרֶב פְּרַעֲה וְכָל הַמּוֹנֵה נָאֵם אֲדֹנָי יְהוִה.	ערל, חרב, נאם	1
אוֹתָם יִרְאֶה פְּרַעֲה וְנָחַם עַל כָּל הַמּוֹנֵה סַלְלֵי חֶרֶב פְּרַעֲה וְכָל חֵילוֹ נָאֵם אֲדֹנָי יְהוִה.	חרב, נאם, כל	1
שָׁמָּה נִסִּיכִי צֶפּוֹן כָּלֵם וְכָל אֲדֹנָי אֲשֶׁר יָרְדוּ אֶת סַלְלִים בְּחַתְּיָתְכֶם מִגְּבוּרַתְכֶם בּוֹשִׁים וַיִּשְׁכְּבוּ עַרְלִים אֶת סַלְלֵי חֶרֶב וַיִּשְׂאוּ כְלִמְתָּם אֶת יָרְדֵי בּוֹר.	ערל, כל, כל	1

Figure 4: In the Hebrew Bible, the lemma **khalal** normally takes on the sense of either **dead body**(as a noun) or **desecrate**(as a verb). This figure presents the appropriate clusters generated by ACCWSI. The “attention highlight” column shows the most relevant context words. In the context of **desecrate** (cluster 0), the attention is paid to words like **God, sacred, nation** etc. while in the context of **dead body** (cluster 1), the attention is paid to **sword, stab, fall**, etc.



origin	joined list of per-word senses	attention_highlight	cluster
P348683:Šamaš eddešū nūr ilī kayyānu Marduk mušim šimāti murrik ūmī kurātī mupahhir ništ saphāti muqerrubū ahāmeš rūgūti atnumma	[] constantly self-renewing light god steady [] one who decrees one who lengthens day short one who gathers people scattered one who brings together far apart one you	šimāti, eddešū, ilu	0
Q004232:EN bēl bēlī šar šarī Šamaš Šamaš dayyān šamē u eršetī bēl māti eddešū nūr ilī kayyānu muštēšer elāti u šaplāti rē ū kīnu ša tenēšēti attā Šamaš dayyānu šīru ša qībīssu lā uttakkaru annašu ilu mamma lā	incantation (label) lord lord king king [] [] judge sky and earth lord land constantly self-renewing light god constant good guide upper world and lower world shepherd true of people you [] judge exalted whose command not be(come) changed approval god anybody not	eddešū, ilu, nūru	0
Q004220:x eddešū nūr ilī kayyān muštēšer elāti u šaplāti rē ū kēnu ša tenēšēti attā Šamaš dayyānu šīru ša qībīssu lā uttakkaru annašu mamma lā	constantly self-renewing light god constant good guide upper world and lower world shepherd true of people you [] judge exalted whose command not be(come) changed approval anybody not	eddešū, ilu, nūru	0
Q005421:tēmēqu Šamaš bēliya ina supēšu ša ilī rabūti puhur mārti ummāni temmēn labiri ippalsūma papāhi u di T ihūto zamar itūrūnim yāti iqbu appalīssa temmēn labiri ša Naram-Sin šar ulla papāhi Šamaš kayyānu mušab ilūtīšu libbī ihdema immeri pānūya papāhi bēlūtīšu u di T ušābbima ina	prayer [] lord through prayer of god great assembly citizen craftsman foundation original discover cella and (deity's) throne-platform check quickly return me say discover foundation original of king of Agade king distant past cella [] permanent residence divinity heart be(come) joyful be(come) bright face cella lordship and (deity's) throne-platform inspect during	hadū, ilūtu, nawāru	1
Q005490:Esagil u Ezida kaqdā kayyānāk asrāt Nabium u Marduk bēliya ašene ' a kayyānam isinātīssu damqātīm akīssu rabītim in gumāhi paglūti alpū šuklūtūti zuluhe damqātīm immer mīru gukkallam ušummu pelā nūnum isūšrum simat appārīm tubik sīrās lā nībi māmiš karānam šattīšam in nuhši u hegallē in maharīssu tēttiq nīsim rapšāti ša	temple of Marduk at Babylon and temple of Nabū at Borsippa constantly constant shrine [] and [] lord constantly seek out constantly festival good (a cultic festival) great with prize bull massive ox perfect (a kind of long-fleeced sheep) good sheep fattening (process) fat-tailed sheep (a rodent) egg fish bird appropriate symbol marsh outpouring (a kind of) beer not number like water wine yearly in abundance and plenty in front parade people extensive that	kaqdā, ešertu, še' ū	1
Q005600:Nabium-kudurri-ušur šar Babili rē ū kīnim migir Marduk iššakku šīri narām Nabu rubām na dam ša alkakāt Marduk bēlu rabi um ilu bānišu u Nabium apilšu kīnim narām šarrūtīšu istene ' ū kayyānam mūda u tele ša irammū puluhtī ilūtīssu ana tēmu ilūtīssu bāsā uznāšu ešru ipēšu ša ana	Nebuchadnezzar II, king of Babylon king Babylon shepherd true favourite [] city-ruler exalted loved one [] prince attentive of way [] lord great god creator and [] heir true loved one kingship constantly seek out constantly one who knows very able one of love reverence divinity for decision divinity existing wisdom wise one expert of for	še' ū, puluhtu, mūdū	1
Q003704:rabēya ašē ' ā asrāt ilāni rabūti šangūti ihūhū irammū nadān zībīya Adad zunnišu umāššera Ea upattira nagbīšu 5 ammat ē išqu ina absinnīšu ērik šubultu 5/6 ammat ešer ebūri napāš Nisaba kayyān ušahnabū gipāru šippāti šummuḫā inbu būlu šutešur ina tālīti ina palēya nuhši tūhdu ina šanātiya kummurū hegallum 10 imēr ē 3	adulthood constantly seek out shrine god great priestly office desire love giving food offering [] rain release [] loosen underground water a unit of length grain be(come) high in furrow be(come) long ear of corn a unit of length success harvest abundance [] constantly make grow luxuriantly pastureland fruit orchard very luxuriant fruit livestock success in offspring during reign (of a king) abundance abundance during year piled up plenty a unit of capacity gram	napāšu, ebūru, šummuḫu	2
Q003707:šeheriya adi rubēya ašē ' ā asrāt ilāni rabūti šangūti ihūhū irammū nadān zībīya Adad zunnišu umāššera Ea upattira nagbīšu 5 ammat ē išqu ina absinnīša ērik šubultu 5/6 ammat ešer ebūri napāš Nisaba kayyān ušahnabū gipāru šippāti šummuḫā inbu būlu šutešur ina tālīti ina palēya nuhši tūhdu ina šanātiya kummurū hegallum 12	childhood to ruler constantly seek out shrine god great priestly office desire love giving food offering [] rain release [] loosen underground water a unit of length grain be(come) high in furrow be(come) long ear of corn a unit of length success harvest abundance [] constantly make grow luxuriantly pastureland fruit orchard very luxuriant fruit livestock success in offspring during reign (of a king) abundance abundance during year piled up plenty	napāšu, ebūru, šummuḫu	2
Q003703:absinnīšu ērik šubultu 5/6 ammat ešer ebūri napāš Nisaba kayyān ušahnabū gipāru šippāti šummuḫā inbu būlu šutešur ina tālīti ina palēya nuhši tūhdu ina šanātiya kummurū hegallu 10 imēr ē 1 imēr karāni BANMIN šamni 1	furrow be(come) long ear of corn a unit of length success harvest abundance [] constantly make grow luxuriantly pastureland fruit orchard very luxuriant fruit livestock success in offspring during reign (of a king) abundance abundance during year piled up plenty a unit of capacity gram a unit of capacity wine oil	napāšu, ebūru, šummuḫu	2
Q003705:ništ āšībūti Qirbit kayyān ihtanabbatū hubut Yamutbali ālu šuātu ina tukulti Aššur Sin Šamaš Bel Nabu Ištār ša Ninua Ištār ša Arba-ili akūd ašlula šallatsu Tandaya itti	people inhabitant [] constantly constantly loot plunder [] city that through help [] [] [] [] [] of Nineveh [] of Arbela conquer carry off booty city ruler of Qirbit with	habātu, nišu, Yamutbal	3
Q006597:tūdāt mātišu ušharrīrma U: x ŠU x x x x alak gerri kayyān ihtanabbatū iprusu x	path land lay waste course road constantly constantly loot block	parāšu, habātu, mātu	3
Q003702:qereb Harehastā lū allik ša Tandaya ana šarrāni abbiya lā iknušu ana nīri u ništ āšībūti Qirbit kayyān ihtanabbatū hubut Yamutbali ālu šuātu	interior [] may go of city ruler of Qirbit to king ancestor not bow down to yoke and people inhabitant [] constantly constantly loot plunder [] city that	habātu, nišu, Yamutbal	3

Figure 6: In the Oracc corpus, containing ancient Akkadian texts, the senses of the lemma **kayyānu** occur in various contexts related to **constant** or **permanent**. This figure presents the four clusters generated by ACCWSI. The “attention highlight” column shows the most relevant context words. The first cluster represents the sense of the stable expression of **self-renewing light god**, the second one means **permanent place of divinity/shrine**, the third one is also a stable expression **constantly make grow**, the fourth is also a stable notion of **permanent loot**. Note that the translation in the **joined list of per-word senses** is just a list of per-word translations that do not necessarily form a coherent sentence or phrase.



origin	joined list of per-word senses	attention_highlight	cluster
P363670:šumma šarru lū šarru x x ēribtu ušaglá x x \$MA x	if king or king (palace-)enterer send into exile	ēribu, šarru, šarru	0
P334718:nišēšūni sahhir dīnaššu ina libbi ūme ša iḡlūni turtānu Zeru-ibni irtugumuniššu issa'ulušu mā issurri ibašši memmēnika rēhe mā memmēniyāma lāšu lā rēhe ūmā atā	people return give in interior day that be(come) deported commander-in-chief [] call (out) ask saying perhaps exist somebody remaining saying anybody (there) is not not remaining now why?	Zeru-ibni, issurri, tartānu	0
P314272:bēlī lū ūda atā ina irti Arihi ubbalūni ušaglanāši u šū iḡṭibi mā Laqaya x	lord may know why? in breast [] bring be(come) deported and he say saying from Laqe	Arihu, wadū, wabālu	0
P334369:Kummaya'e dayālī ša issu Kumme ana nāgurtu illikūni udīni lā illakūni ammākamma šunu šarru bēlī liš'al luššiši issurri qanni ammūte ušagalūšunu šarru bēlī ina	Kummean scout who from [] to hire come yet not come there they king lord ask investigate perhaps together with that be(come) deported king lord to	qannu, issurri, uššūšu	0
P237270:ana šar mātāti bēlīya aradka Kudurru Aššur Šamaš Bel u Nabu ana šar mātāti bēlīya likrubū ultu muhhi ūmu ša šarru bēlā ušeglanni šabtāk u ašbāk ūmussu ana šarri bēlīya ušalla Nabu-killanni kī	to king land lord servant [] [] [] [] and [] to king land lord bless since skull day that king lord be(come) deported kept in confinement and sitting daily to king lord pray to [] when	ūmussu, ašbu, šarru	0
P334504:nušagli o issurri šarru bēlī iḡabbi mā ša bēt Ilumma-taklak Lahiraya šunu x x x	be(come) deported perhaps king lord say saying of house [] Lahiraean they	issurri, šarru, qabū	0
P237257:x x x x akī hītu ša ana šarri bēlīya iḡṭu ultegeli u	in accordance with crime which against king lord sin deport and	šarru, haṭū, hītu	0

Figure 7: ACCWSI produced a single cluster for some instances of the **galū** lemma, while resolving much of the results without any cluster. Some improvements should be applied to ACCWSI in order to detect subtle nuances, controversial even for humans. Note that the translation in the **joined list of per-word senses** is just a list of per-word translations that do not necessarily form a coherent sentence or phrase.

## ACKNOWLEDGMENTS

This research is supported by the Ministry of Science & Technology, Israel, Grant 3-16464.

## References

- Eneko Agirre and German Rigau. Word sense disambiguation using conceptual density. *arXiv preprint cmp-1g/9606007*, 1996.
- Asaf Amrami and Yoav Goldberg. Towards better substitution-based word sense induction. *arXiv preprint arXiv:1905.12598*, 2019.
- Stefan Bordag. Word sense induction: Triplet-based clustering and automatic evaluation. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, 2006.
- Baobao Chang, Wenzhe Pei, and Miaohong Chen. Inducing word sense with automatically learned hidden concepts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 355–364, 2014.
- Jim Cowie, Joe Guthrie, and Louise Guthrie. Lexical disambiguation using simulated annealing. In *COLING 1992 Volume 1: The 14th International Conference on Computational Linguistics*, 1992.
- Matan Eyal, Shoval Sadde, Hillel Taub-Tabib, and Yoav Goldberg. Large scale substitution-based word sense induction. *arXiv preprint arXiv:2110.07681*, 2021.
- Aysu Ezen-Can. A comparison of lstm and bert for small corpus. *arXiv preprint arXiv:2009.05451*, 2020.
- Evelina Fedorenko and Rosemary Varley. Language and thought are not the same thing: evidence from neuroimaging and neurological patients. *Annals of the New York Academy of Sciences*, 1369(1):132, 2016.
- Andrea Galassi, Marco Lippi, and Paolo Torrioni. Attention in natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.
- Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.’s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.
- Mika Hämmäläinen and Khalid Alnajjar. Let’s FACE it. Finnish poetry generation with aesthetics and framing. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 290–300, Tokyo, Japan, October–November 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-8637. URL <https://aclanthology.org/W19-8637>.
- Greg Hamerly and Charles Elkan. Learning the k in k-means. *Advances in neural information processing systems*, 16:281–288, 2004.
- Dekang Lin. Automatic retrieval and clustering of similar words. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 2*, pages 768–774, 1998.
- Suresh Manandhar, Ioannis Klapaftis, Dmitriy Dligach, and Sameer Pradhan. Semeval-2010 task 14: Word sense induction & disambiguation. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 63–68, 2010.
- Patrick Pantel and Dekang Lin. Discovering word senses from text. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 613–619, 2002.
- Douglas A Reynolds. Gaussian mixture models. *Encyclopedia of biometrics*, 741:659–663, 2009.
- Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21, 2017.
- Hinrich Schütze. Automatic word sense discrimination. *Computational linguistics*, 24(1):97–123, 1998.
- Goldee Udani, Shachi Dave, Anthony Davis, and Tim Sibley. Noun sense induction using web search results. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 657–658, 2005.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.