Deep-change at AXOLOTL-24: Orchestrating WSD and WSI Models for Semantic Change Modeling

Denis Kokosinskii[1,2], Mikhail Kuklin[1,3], and Nikolay Arefyev[4]

[1]Moscow State University, Russia
[2]SaluteDevices, Russia
[3]Yandex, Russia
[4]University of Oslo, Norway
kokosinskiidv@my.msu.ru, kuklin.mike@yandex.ru, nikolare@uio.no

Abstract

This paper describes our solution of the first subtask from the AXOLOTL-24 shared task on Semantic Change Modeling. The goal of this subtask is to distribute a given set of usages of a polysemous word from a newer time period between senses of this word from an older time period and clusters representing gained senses of this word. We propose and experiment with three new methods solving this task. Our methods achieve SOTA results according to both official metrics of the first substask. Additionally, we develop a model that can tell if a given word usage is not described by any of the provided sense definitions. This model serves as a component in one of our methods, but can potentially be useful on its own.

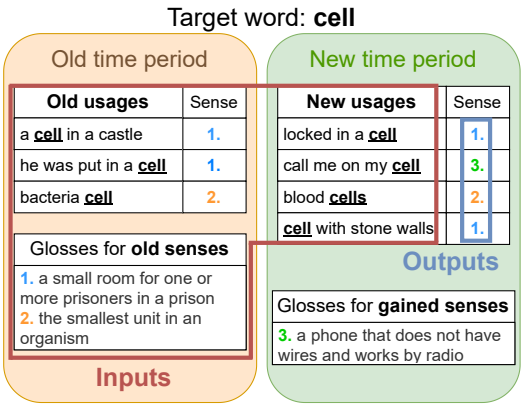

Figure 1: An example of data for the first subtask of AXOLOTL-24.

1 Introduction

The shared task on explainable Semantic Change Modeling (SCM) AXOLOTL-24 (Fedorova et al., 2024) is related to automation of Lexical Semantic Change (LSC) studies, i.e. linguistic studies on how word meanings change over time. It consists of two subtasks, however, we focus on the first one and skip the definition generation subtask. Unlike other shared tasks LSC held before, the first subtask of AXOLOTL-24 requires automatic annotation of individual usages of target words instead of target words as a whole. An example of the provided data and required outputs is shown on Figure 1. Namely, for each target word, two sets of usages from an older and a newer period are given (we will call them *old* and *new* usages). Additionally, a set of glosses describing word senses in the older time period (*old senses*) are provided, and the old usages are annotated with these sense glosses. Senses occurring among the new usages (*new senses*) should be discovered automatically. To be precise, the goal is to annotate each new usage with one of the given old sense glosses, or a unique sense identifier if none of them is applicable. We will refer to those senses that occur only among old and only among new usages as *lost* and *gained* senses, and all other senses as *stable* senses.

To solve the task, we experiment with three types of models. Word Sense Disambiguation (WSD) models for a given word usage select among given glosses the most suitable one. Word Sense Induction (WSI) models group word usages into clusters corresponding to word senses, they are applicable even when sense descriptions are not available. Finally, Novel Sense Detection (NSD) models find usages corresponding to unknown word senses, the ones that are not covered by the provided definitions. We propose three methods that solve the task. Our best solution denoted as Outlier2Cluster combines all three types of models in a novel way, essentially using an NSD model for each usage to decide whether to return a definition selected by a WSD model, or an identifier of a cluster this usage was put into by a WSI model. On average across languages, this solution achieves SOTA results among all participants of the first substask of AXOLOTL-24 according to both official metrics.

An important additional contribution is the pro-

Proceedings of the 5th Workshop on Computational Approaches to Historical Language Change, pages 168–179

posed NSD model and the related experiments. We study the importance of different features of the NSD model and its effect on SCM quality. Our experiments suggest that improving NSD quality is the most promising direction for the future.

2 Related work

LSCD methods. Several shared tasks related to LSCD were organized in the past, including Schlechtweg et al. (2020); Kutuzov and Pivovarova (2021); Zamora-Reina et al. (2022). Unlike AXOLOTL-24 (Fedorova et al., 2024), they required word-level predictions from their participants, either in the form of word ranking or binary word classification. This type of task setup is generally mentioned under the name Lexical Semantic Change Detection / Discovery (LSCD). In the earlier shared tasks the best results were achieved by solutions that employed non-contextualized word-level embeddings such as word2vec (Mikolov et al., 2013) and vector alignment methods such as Canonical Correlation Analysis and Orthogonal Procrustes Alignment (Pömsl and Lyapin, 2020; Pražák et al., 2020). However, recently token-level methods (Laicher et al., 2021; Rachinskiy and Arefyev, 2021a, 2022) have surpassed them. These methods rely on masked language models fine-tuned on existing datasets for various tasks of lexical semantics. For instance, solutions relying on the contextualized embeddings from GlossReader, which is a WSD system, have shown SOTA results in the shared tasks on LSCD in Russian and Spanish (Rachinskiy and Arefyev, 2021a, 2022). Methods proposed in this work exploit GlossReader too, both as a WSD model and as a source of contextualized embeddings well-suited for LSC-related tasks.

GlossReader is a multilingual gloss-based WSD model originally developed to solve the Word-in-Context task (Rachinskiy and Arefyev, 2021b). It modifies the English WSD model BEM (Blevins and Zettlemoyer, 2020) replacing the backbone with the multilingual XLM-R language model (Conneau et al., 2020). The model consists of a gloss encoder and a context encoder, both initialized with the XLM-R weights and fine-tuned jointly learning to select among all glosses of a target word the one describing its sense in a given context. Specifically, the dot product between the context embedding and the correct gloss embedding is maximized.

NSD methods. Several methods were proposed to solve the NSD task. Some of them perform WSI internally. For instance, Lau et al. (2012); Cook et al. (2014) employ a topic modelling approach to jointly cluster old and new usages using the Hierarchical Dirichlet Process. Clusters are ranked based on the novelty score (the difference between estimated probabilities of a cluster appearing in the new and the old corpus). While the method was originally designed for LSCD, the novelty ranking of senses can be combined with a static threshold to identify novel senses.

Alternatively, Mitra et al. (2015) performs WSI separately for an old and a new corpus on graphs, where an edge weight between two words is proportional to the number of words appearing in bigrams with both of them. A cluster in the new corpus is labeled as a novel sense if words in this cluster have weak links with the target word in the graph for the old corpus. A recent method by Ma et al. (2024) uses BERT (Devlin et al., 2019) to build contextualized representations. It employs agglomerative clustering to perform WSI and then matches old and new clusters based on their centroids. The new clusters that are not matched are considered novel senses. Similarly to this method we use agglomerative clustering for WSI, but employing GlossReader to obtain contextualized embeddings.

In Erk (2006) several NSD methods were proposed to detect word senses that are not described in FrameNet (Baker et al., 1998). Instead of relying on WSI, similarly to our NSD method their best method formulates the task as an outlier detection problem. They employ distances between old and new usages requiring a significant number of old usages for each sense, which are not always available in AXOLOTL-24. Thus, we rely on distances between new usages and old glosses instead. Another similar method is introduced in Lautenschlager et al. (2024). They use the XL-LEXEME model (Cassotti et al., 2023) to build representations for usages and senses. Sense representations are built from glosses or example usages of senses taken from dictionaries. They do not always contain the target word, which makes application of XL-LEXEME non-trivial. Authors attempt to solve this problem by modifying glosses and example usages to include the target word. For each usage its nearest sense is found based on the cosine similarity or the Spearman's correlation between their embeddings. If the similarity is above a threshold, the usage is considered to belong to some non-

described sense. Our methods also rely on usage and sense representations, but we use GlossReader which has a separate gloss encoder and does not require any preprocessing for glosses. We experiment with many measures of similarity between a sense and a usage embedding, and found the manhattan distance between l1-normalized embeddings to outperform other measures and a classifier on a combination of measures to perform best. However, we did not experiment with example usages from sense inventories. When such usages are available, being combined with glosses they may potentially improve sense representations.

3 Methods

3.1 Target word positions

All our methods assume that a usage is represented as a string and two character-level indices pointing to a target word occurrence inside this string. However, for the Russian subsets these indices were absent. To find them, we first generated all grammatical forms for each target lemma using Pymorphy2 (Korobov, 2015). Then retrieved all occurrences of these forms as separate tokens in the provided usages employing regular expressions.[1] For usages with several occurrences of the target word we selected one of them that has both left and right context of reasonable length.[2] We inspected new usages from the development and the test sets that did not contain any of the automatically generated word forms and added absent forms manually, then reran retrieval.[3]

3.2 WSD methods

The first group of methods in our experiments include pure WSD methods, which select one of the provided definitions of old senses for each new usage, and thus, cannot discover gained senses.

GlossReader. We employ the original GlossReader model (Rachinskiy and Arefyev, 2021b) as the baseline. For a given **new** usage u of a target word w its usage representation r_u is built with the context encoder. Then gloss representations r_g are built for each gloss g of the target word w using the gloss encoder. Finally, the gloss with the highest dot product similarity to the usage is selected.

To improve the results, we further fine-tune the GlossReader model on the data of AXOLOTL-24.

GlossReader FiEnRu is fine-tuned following the original GlossReader training procedure on three datasets: the train sets of the shared task in Finnish and Russian, and the English WSD dataset SemCor (Miller et al., 1994) which GlossReader was originally trained on. We employ all old and new usages from the Russian and Finnish datasets along with their sense definitions. We fined-tuned for 3 epochs using 90/10% train/validation split to select the best checkpoint.[4]

GlossReader Ru is fine-tuned exactly the same way, but only on the train set in Russian.

GlossReader Fi SG is fine-tuned on the Finnish train set only. Unlike two previous models, we made an attempt to teach this model how to discover novel senses. Specifically, we replaced all glosses of gained senses with a Special Gloss (SG) "the sense of the word is unknown" in Finnish[5] and fine-tuned the model as before. For inference we tried adding the special gloss to the provided old glosses, essentially extending the WSD model with NSD abilities. However, this resulted in a noticeable decrease of the metrics on the Finnish development set. Thus, we decided to use the special gloss for training only.[6]

3.3 WSI methods

Unlike WSD methods, WSI methods do not use definitions or any other descriptions of word senses. Instead they discover senses of a word from an unlabeled set of its usages by splitting this set into clusters hopefully corresponding to word senses. WSI methods cannot attribute usages to the provided old glosses, but can potentially group usages

[1] E.g. `'\b(cat|cats)\b'`, where `\b` denotes a word boundary. Matching is case-insensitive.

[2] This idea is based on our observations that a word occurrence is encoded sub-optimally when it is either the first or the last token, which is probably related to confusion of Transformer heads that have learnt to attend to the adjacent tokens (Voita et al., 2019). The heuristic implemented takes the second to last occurrence if there are more than two of them. For two occurrences it takes $argmax_{u \in \{u1,u2\}} min(l_u, r_u)$, where l_u, r_u are the lengths of the left and the right contexts.

[3] Repeating this manual procedure for all Russian data requires significantly more efforts and would have few benefits for our methods. Thus, all old usages having this issue were left without indices and new usages from the training set we dropped.

[4] The last checkpoint was selected, though after ≈0.5 epochs metrics improve very slowly.

[5] "sanan merkitystä ei tunneta" as translated by Google Translate

[6] The majority of words in the Finnish dataset have one sense only, see Section 4.2. Pure WSD methods always return perfect predictions for such cases, thus, it is very hard to compete with them on this dataset. In the future we plan to experiment with this model on the Russian dataset having much smaller proportion of such words.

of the same sense, including gained senses, into a separate cluster.

Agglomerative is the only WSI method we propose and experiment with. For each new usage its representation r_u is built using the context encoder of the original GlossReader model. Then we perform agglomerative clustering of old usages using the cosine distance and average linkage on these representations. This clustering algorithm was successfully used to cluster vectors of lexical substitutes, another kind of word sense representations, in several substitution-based WSI methods (Amrami and Goldberg, 2018, 2019; Arefyev et al., 2020; Kokosinskii and Arefyev, 2024), as well as for LSCD (Laicher et al., 2021; Ma et al., 2024).

Agglomerative clustering starts with each usage in a separate cluster, then iteratively merges two closest clusters. The distance between two clusters is the average pairwise cosine distance from the usages in the first cluster to the usages in the second one. Merging stops when the predefined number of clusters is reached. We range the number of clusters between 2 and 9 and select a clustering with the highest Calinski-Harabasz score (Caliński and Harabasz, 1974).[7]

3.4 SCM methods

WSD and WSI methods provide only partial solutions of the semantic change modeling task, the former cannot discover novel senses, and the latter cannot annotate usages with the old glosses provided. We propose three new methods developed to fully solve the task.

3.4.1 AggloM

Our first SCM method modifies the Agglomerative WSI method by incorporating old usages and senses into the clustering process. We perform agglomerative clustering of a set containing both old and new usages of a target word. Initially, each new usage is assigned to a separate cluster. The old usages are clustered according to the provided sense annotations. Then at each iteration we compute the distances from each cluster containing only new usages to all other clusters. The distance between two clusters is defined as the minimum cosine distance between the usage representations from the first and the second cluster.[8] We then merge two nearest clusters, one of which contains new usages only. This iterative merging process stops when the number of clusters is larger than the number of old senses by $k \geq 0$. Therefore AggloM returns exactly k novel senses, where k is a hyperparameter.[9] We do not use this method on the Russian datasets because for most senses there are no old usages there.

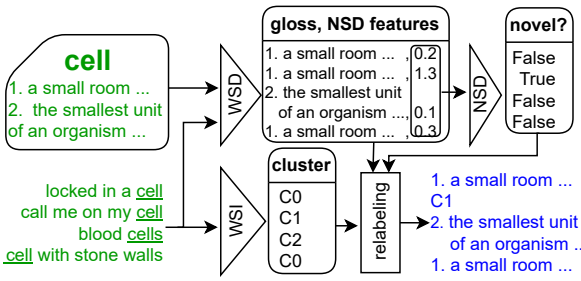

Figure 2: Outlier2Cluster pipeline. Inputs are in green and outputs are in blue. Triangles denote ML models.

AggloM FiEnRu is identical to AggloM but relies on the fine-tuned GlossReader FiEnRu.

3.4.2 Cluster2sense

In the second SCM method we first independently cluster new usages using the Agglomerative WSI method and annotate them with the old senses using GlossReader FiEnRu. We then keep the clustering obtained from WSI, but relabel those clusters that overlap heavily with one of the predicted senses. Specifically, we label a cluster c with a sense s if c has the highest Jaccard similarity to s among all the old senses of the target word, and at the same time s has the highest similarity to c among all the clusters built for new usages of this word. Notably, two clusters cannot be labeled with a single sense, thus the clustering of usages is identical to the one originally predicted by WSI. Some clusters will not be labeled with any sense, thus, Cluster2sense can discover gained senses. At the same time, some senses will not be assigned to any cluster, which means the potential to discover lost senses as well.

3.4.3 Outlier2Cluster

Unlike Cluster2Sense which relabels whole clusters, Outlier2Cluster relabels individual usages. Figure 2 shows the processing pipeline. First WSD and WSI predictions are independently made

[7] For one or two usages the Calinski-Harabasz score is not defined. We return a single cluster in such cases.

[8] This is known as single linkage.

[9] In the preliminary experiments on the Finnish development set we selected $k = 0$, which means that all new usages are eventually merged into clusters representing old senses. This is likely related to the low proportion of gained senses in this dataset and noisy usages which make them hard to discover.

by GlossReader FiEnRu and Agglomerative respectively. Then we discover usages of gained senses. For that we propose a Novel Sense Detection (NSD) model finding usages of those senses that we do not have definitions for.[10] Finally, we return WSI predictions for all these discovered usages, and WSD predictions for all other usages.

Novel sense detection. We treat the NSD task as an outlier detection problem, essentially finding those usages that are distant enough from all the provided definitions. Since GlossReader selects the most similar definition for a given usage, it is enough to check if this definition is distant enough to conclude that the usage is an outlier. To check this we employ a logistic regression classifier. Each input example corresponds to a single usage and a gloss selected for this usage by the WSD model. The output is 1 if this usage is an outlier, i.e. does not belong to the predicted sense, and 0 otherwise.

We use distances (computed with several distance functions) between GlossReader representations of the new usages and the glosses for old senses as features for logistic regression.

For the new usage u and the selected definition g we take the corresponding representations r_u and r_g from a gloss encoder and a context encoder respectively. We take these representations from two different GlossReader models, the original one and GlossReader FiEnRu, and calculate distances from r_u to r_g using different distance and normalization functions. This gives 10 different features presented in Table 1. We also include three extra features: the number of old usages, old senses, and new usages for the target word in the dataset. We employ the Standard Scaler to normalize features and train the logistic regression with L2 regularization of $C = 1$.

Thus, the trained logistic regression can be used for each usage to decide whether the WSD method has assigned a correct sense or should be replaced with some cluster corresponding to a gained sense. If the score is above a threshold of 0.65, which was selected on the development sets of the shared task, the usage is considered an outlier.

We train two NSD models on the Russian and the Finnish development sets separately and use the trained models for the corresponding test sets.

[10] In the context of the shared task these are gained senses. However, the approach is general enough to discover lost senses when a modern dictionary and old usages are given, or just senses from the same time period as the dictionary but not covered by it.

	Distance Function	Cos.	Euclid.	Manh.
Encoders	**Normalized**			
GR FiEnRu	No	✓	✓	✓
GR FiEnRu	L1-norm			✓
GR FiEnRu	L2-norm		✓	
GR	No	✓	✓	✓
GR	L1-norm			✓
GR	L2-norm		✓	

Table 1: Ten distance-based features used in the NSD model. Distances are calculated between usage and gloss representations obtained from context and gloss encoders of the same GlossReader model. GR stand for GlossReader, Cos. is the cosine distance, Euclid. is the euclidean distance, Manh. is the manhattan distance.

For the surprise language, we do not have labeled data to select one of two models or train a separate model, thus, we simply report the results of both models.

Outlier relabeling. We experiment with two ways of assigning clusters to the detected outliers. Our first approach (**w/o WSI**) groups all outliers into a single new cluster. Alternatively, **w/ WSI** approach assigns the clusters predicted by the WSI method to outliers. We use the first option for the Finnish test set, as we observed that the words in the Finnish development set rarely have more than one gained sense. On the contrary, the words in the Russian development set have many gained senses, therefore, we employ w/ WSI for the Russian test set. For the surprise language Outlier2Cluster$_{fi}$ employs w/o WSI and Outlier2Cluster^{ru} employs w/ WSI.

All of the described methods are briefly summarized in Table 2.

4 Evaluation setup

The first AXOLOTL-24 subtask evaluates semantic change modeling systems in three diachronic datasets in Finnish, Russian, and German (Fedorova et al., 2024). Train and development sets are provided for the first two, but not for the last. We will now describe the datasets in more detail.

4.1 Data sources

The source for the Finnish dataset of the shared task is resource (1997). The usages are divided into two groups: before 1700 and after 1700. The usages in the dataset are not complete sentences but short phrases. Some parts of the phrase can be missing and replaced with double hyphens, presumably due to OCR errors. Furthermore, the usages from both the old and the new corpus exhibit no-

	Underlying embeddings	Requires usages of old senses	Requires old glosses	Requires a train set with gained senses*	Able to discover gained senses	Able to predict old senses
GR	GR	-	✓	-	-	✓
GR FiEnRu	GR FiEnRu	-	✓	-	-	✓
GR Ru	GR Ru	-	✓	-	-	✓
GR Fi SG	GR Fi SG	-	✓	✓	-	✓
Agglomerative	GR	-	-	-	✓	-
AggloM	GR	✓	-	-	if $k > 0$	✓
AggloM FiEnRu	GR FiEnRu	✓	-	-	if $k > 0$	✓
Cluster2Sense	GR, GR FiEnRu	-	✓	-	✓	✓
Outlier2Cluster	GR, GR FiEnRu	-	✓	✓	✓	✓

Table 2: A brief description of the proposed methods. GR stands for GlossReader model. *GR Fi SG is trained to predict the special gloss for usages of all gained senses. In Outlier2Cluster the NSD model is trained to detect usages of gained sense.

table differences from modern Finnish. They often feature characters (such as c, z, w, and x), that are not commonly found in contemporary Finnish. It is important to highlight that the glosses provided for word senses are in modern Finnish.

Two data sources used to create the Russian dataset are Dahl (1909) processed by Mikhaylov and Shershneva (2019) and Mickus et al. (2022). The first one was the source of old usages and glosses, and the latter provided new usages and glosses. However, the specific procedure used to map senses between these two sources was undisclosed at the time of the competition. Some old senses are not accompanied by old usages in the Russian datasets. Consequently, our methods for the Russian datasets do not rely on the old usages. Notably, the Russian datasets lack information regarding the position of a target word within a usage or the actual word form of the target word. As a result, we incorporate the identification of the target word's position within a usage as a preprocessing step in our solution.

The shared task also includes a test dataset in a surprise language revealed only at the test phase of competition with no development or train sets. The source of this dataset is a German diachronic corpus with sense annotations (Schlechtweg et al., 2020; Schlechtweg, 2023).

4.2 Data Statistics

To get insights into the data we categorize the target words within the train and the development sets based on several characteristics:

- Has lost senses: does the word have old senses for which there are no new usage?
- Number of gained senses: how many senses are there having new usages only?

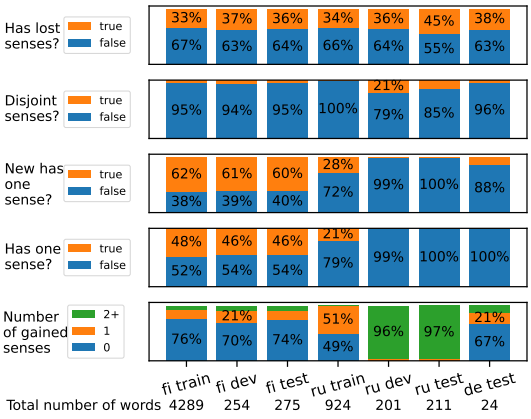

Figure 3: Proportions of target words falling into different categories in the shared task datasets.

- Disjoint senses: are the sets of senses for old and new examples disjoint?
- New has one sense: do all the new usages have the same meaning?
- Has one sense: do all the usages (both old and new) have the same meaning?

The number of target words in each category for all[11] the datasets of the shared task is presented on Figure 3.

In the Finnish datasets, almost half of target words have only one sense and approximately 70% of words have no gained senses. Therefore, the conservative methods that rarely discover gained senses are preferable for the Finnish datasets.

The main observation for the Russian datasets is the dramatic differences in proportions of almost all categories between the train and the development

[11]This information for the test sets was not available during the competition.

set. We can see that the statistics of the test set are similar to those of the development set. Contrary to the Finnish sets, almost all words in the Russian development set have gained senses. Therefore, methods which are prone to predict new senses rather than old ones are preferable for the Russian development set.

The German dataset is relatively small and contains 8 times fewer words than the other test sets. We can see that it is similar to the Finnish datasets in the proportion of gained and lost senses.

4.3 Metrics

The shared task employs two metrics to evaluate the systems, the Adjusted Rand Index (ARI) and the F1 score.

ARI (Hubert and Arabie, 1985) is a well-established clustering metric employed to evaluate how well new usages are clustered by a system. In the subtask, ARI is computed for all the new usages of a target word, the ground truth clusters correspond to senses. Notably, cluster labels are not taken into account by ARI. It means that old senses and gained senses are indistinguishable from each other in terms of ARI.

The F1 score is used in the first subtask to estimate how well a system can discriminate between old senses. It is computed only for the new usages of the old senses, and not for the usages of the gained senses. The F1 score for a target word is the average of the F1 scores for all old senses. If a target word does not have any new usages with the old senses, it is arbitrarily assigned the F1 score of 1 if old senses are not predicted for any of its usages and 0 otherwise. Thus, in this edge case a system is heavily penalized when even a single usage is misclassified as one of the old senses.

All new usages of the old senses which are (incorrectly) predicted as belonging to a gained sense are considered to belong to a single auxiliary "novel" class when calculating the F1 score. The F1 score for this class is zero as it has zero precision. For this reason, even a single usage misclassified as a gained sense can dramatically affect the overall score for a target word independently of the total number of its usages.[12]

[12]Assume the target word has k old senses. In case when only old senses are predicted: $F = \frac{F_1+\ldots+F_k}{k}$. If we replace one of the correct predictions of sense 1 with an incorrect prediction of a gained sense: $F' = \frac{F'_1+\ldots+F_k\mathbf{+0}}{\mathbf{k+1}} < \frac{F_1+\ldots+F_k\mathbf{+0}}{\mathbf{k+1}}$. The drop in this metric is $\frac{F}{F'} > \frac{k+1}{k}$ E.g. in the case $k = 1$, which is a frequent case in the Finnish AXOLOTL-24 dataset, an incorrect prediction of a gained sense for a single usage results in more than 2x decrease of the F1 score.

5 Results

5.1 Our submissions

The number of submissions for the test sets per team was not limited in the competition. We evaluate ten models on the test sets: four WSD models (based on GlossReader, GlossReader FiEnRu, GlossReader Ru, and GlossReader Fi SG), one WSI model (Agglomerative with GlossReader representations), two AggloM models (based on GlossReader and GlossReader FiEnRu representations), Cluster2Sense, and Outlier2Cluster with different configurations for the German dataset: Outlier2Cluster^{ru} and Outlier2Cluster$_{fi}$. Table 3 demonstrates the evaluation results. We also include the best submissions from other teams for comparison.

WSD and WSI. The best results in terms of the F1 score are achieved by pure WSD methods. The F1 score is calculated only for the usages of old senses, this gives a huge advantage to WSD methods because incorrect prediction of old senses for usages of gained senses is not penalized, while the opposite reduces the F1 score severely as explained in Section 4.3.

WSD methods have notably higher ARI than Agglomerative and Cluster2Sense (both of them predict the same clusters but label them differently) for the Finnish and German datasets. On the contrary, Agglomerative and Cluster2Sense are the best-performing methods for the Russian dataset. Our explanation for this fact comes from the analysis in Section 4.2. The sets of senses of the new and the old usages in the Finnish and German datasets overlap heavily, which is beneficial for WSD methods. The overlap is much smaller for the Russian dataset, which hurts ARI of the WSD methods. Discovering gained senses is crucial for the Russian dev and test set.

AggloM. The AggloM method with the hyperparameter $k = 0$ (never predicts gained senses) does not fall far behind pure WSD methods. The main reasons for that probably are the usage of the same underlying context encoder and prediction of only old senses. Therefore, AggloM is a viable alternative to the GlossReader models when word senses are described with usage examples instead of sense definitions.

Outlier2Cluster. Outlier2Cluster achieves

Method	ARI Fi	ARI Ru	ARI De	ARI FiRu	ARI AVG	F1 Fi	F1 Ru	F1 De	F1 FiRu	F1 AVG
WSD methods										
GR	0.581	0.041	0.386	0.311	0.336	0.690	◇0.721	0.694	0.706	0.702
GR FiEnRu	◇**0.649**	0.048	◇0.521	**0.348**	0.406	◇**0.756**	◇**0.750**	◇0.745	◇**0.753**	◇**0.750**
GR Ru	0.568	0.053	0.464	0.310	0.361	0.568	◇**0.750**	◇0.724	0.659	0.681
GR Fi SG	◇0.638	**0.059**	◇**0.543**	**0.348**	**0.413**	◇0.752	◇0.729	◇**0.758**	◇0.741	◇0.746
WSI methods										
Agglomerative	**0.209**	◇**0.259**	**0.316**	**0.234**	**0.261**	**0.055**	**0.152**	**0.042**	**0.104**	**0.083**
SCM methods										
AggloM	0.581	0	**0.492**	0.290	0.357	0.674	0	0.695	0.337	0.456
AggloM FiEnRu	◇0.631	0	0.485	0.315	0.372	◇0.731	0	0.639	0.366	0.457
Cluster2Sense	0.209	◇**0.259**	0.316	0.234	0.261	0.432	0.346	0.432	0.389	0.403
Outlier2Cluster $^{ru}_{fi}$	◇**0.649**	◇0.247	*0.322* *0.480*	◇**0.448**	*0.406* ◇***0.459***	◇**0.756**	**0.645**	*0.510* ◇***0.745***	**0.701**	*0.637* ◇***0.715***
Other teams										
Holotniekat	**0.596**	0.043	0.298	**0.319**	**0.312**	**0.655**	**0.661**	0.608	**0.658**	**0.641**
TartuNLP	0.437	0.098	**0.396**	0.267	0.310	0.550	0.640	0.580	0.595	0.590
IMS_Stuttgart	0.548	0	0.314	0.274	0.287	0.590	0.570	0.300	0.580	0.487
ABDN-NLP	0.553	0.009	0.102	0.281	0.221	**0.655**	0	**0.638**	0.328	0.431
WooperNLP	0.428	**0.132**	0	0.280	0.186	0.503	0.446	0	0.475	0.316
Baseline	0.023	0.079	0.022	0.051	0.041	0.230	0.260	0.130	0.245	0.207

Table 3: The results on the test tests. The best result for each metric is underlined, the best result in each group is in **bold font**. A diamond (◇) denotes those results that are worse than the best one, but the difference is practically insignificant (we consider relative differences smaller than 0.05 as practically insignificant). The official AXOLOTL-24 leaderboard is based on the average metrics across the languages having the training sets provided (the FiRu columns) and all languages (the AVG columns).

SOTA or near-SOTA ARI[13] for Russian and Finnish, but falls behind WSD methods for German, which has no labeled data to train a dedicated NSD model. However, Outlier2Cluster can discover gained senses unlike WSD methods. Thus, we consider Outlier2Cluster to be preferable for the SCM task and suggest training the NSD model for each language of interest.[14]

The important hyperparameter of the NSD model, and consequently the Outlier2Cluster model exploiting it as a component, is the threshold dividing usages into outliers and normal usages. Figure 4 shows the dependence of the metrics on the threshold value for the Finnish and Russian development sets. Both w/ WSI and w/o WSI versions of Outlier2cluster are included. We also compute the results of Outlier2Cluster with the WSI oracle which perfectly clusters the detected outliers according to their ground truth senses, and the NSD oracle which perfectly detects usages of gained senses. The methods we study in this Section are briefly summarized in Table 4.

We can see that the F1 score (computed only

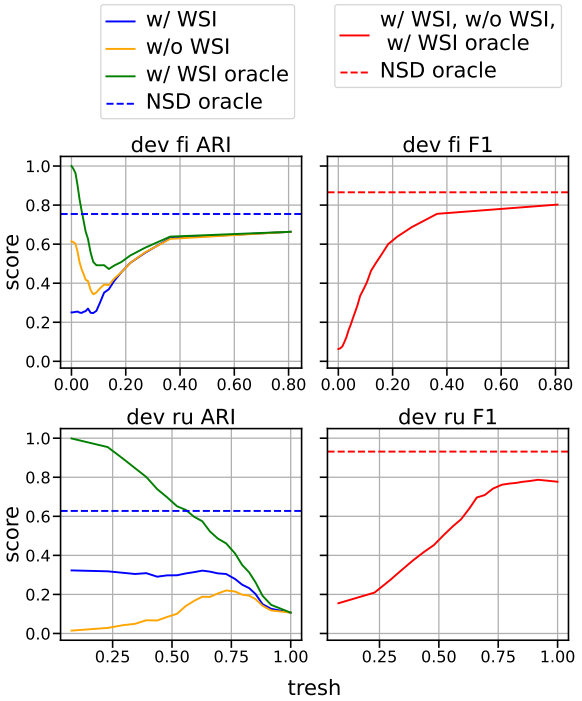

Figure 4: ARI and F1 on the development sets depending on the threshold of novel sense detector. Higher threshold means higher proportion of WSD predictions and less WSI predictions.

[13]We made Outlier2Cluster$_{fi}$ submissions in the competition separately for different datasets. For this reason, it was not selected as our best submission by the competition organizers.

[14]We used only small development sets with ≈ 200 target words to train novel sense detection models.

Method	WSD	WSI	NSD
w/ WSI	GR FiEnRu	Agglomer.	LogReg
w/o WSI	GR FiEnRu	One cluster	LogReg
w/ WSI oracle	GR FiEnRu	Oracle	LogReg
NSD oracle	GR FiEnRu	Agglomer.	Oracle

Table 4: A brief summary of methods for the NSD threshold study.

over new usages with old senses) monotonically increases with the increasing threshold, i.e. with fewer outliers detected. This again shows that trying to detect usages of gained senses and clean the old senses from them hurts the F1 score, supporting the criticism of this metric in Section 4.3.

ARI reaches a peak at the threshold of 0.65 for the Russian dataset with F1 being close to maximum as well. We therefore set the threshold at 0.65 for the Russian NSD model. This gives the SCM model that almost achieves the ARI of pure WSI predictions (threshold of 0) while having only a bit smaller F1 score compared to the best WSD model.

For the Finnish dataset, higher ARI monotonically increase with the threshold, i.e. with the proportion of predictions taken from the WSD model. This agrees with the observations from Table 3 that the pure WSD models give the best ARI for Finnish. We can also see that the threshold values in the middle, where neither WSI nor WSD predictions are dominant, result in a significant decrease in ARI. It means, that our NSD model cannot be used effectively to combine the predictions for Finnish. We select a high threshold of 0.65 for the Finnish dataset, resulting in a low number of outliers. Consequently, the novel sense detector predicts less than 1% of usages to be outliers in the Finnish test set, compared to 42% of usages predicted as outliers for the Russian test dataset.

We can observe that according to the F1 score, the NSD oracle performs better than the pure WSD method, especially on the Russian development set. The reason lies in the words with disjointed senses. Since there are no new usages of old senses for such words, the ordinary F1 score and it is arbitrarily defined as 1 if all usages are recognized as usages of gained senses, i.e. put into new clusters, and 0 otherwise. Thus, the ideal processing of these edge cases is crucial for the F1 score, but can hardly be achieved unless the NSD oracle is employed. For other words it does not help. Considering ARI, the NSD oracle performs much better than w/WSI on the Russian dataset. It means that better NSD models may help greatly improve clustering.

According to the results of w/ WSI oracle on the Finnish development set, it is impossible to increase ARI with better WSI method without a huge drop in the F1 score. For the Russian dataset situation is the opposite. The main reason is likely the average number of gained senses per word in these datasets as described in Section 4.2. Only 7% of words in the Finnish dataset have gained two or more senses, therefore the perfect clustering of the gained senses does not increase the results significantly compared to merging all gained senses into a single cluster. On the contrary, 97% of the word in the Russian have two or more gained senses, making WSI necessary.

6 Conclusion

We have proposed three new methods that solve the SCM task. Our solution achieves SOTA results among all participants of the first subtask of the AXOLOTL-24 shared task. Additional experiments propose directions of further improvement of the developed models, NSD being potentially the most promising one.

7 Limitations

While our methods can in theory be applied to any SCM dataset, we acknowledge that they may be overspecified for the first subtask of AXOLOTL-24. Notably, we extensively use the train sets provided for the competition in Finnish and Russian to train the embedding model and to optimize the hyperparameters. While we also evaluate on the German dataset in a zero-shot fashion, the results may be unreliable due to relatively small size of the dataset.

Semantic change modeling may be of particular interest in studies of older time periods, where the language is quite different from its modern state. The underlying model, GlossReader, is a finetuned version of XLM-R, which was not specifically designed to handle old languages. In this case dataset-specific finetuning of the base GlossReader may become even more relevant.

8 Acknowledgements

Nikolay Arefyev has received funding from the European Union's Horizon Europe research and innovation program under Grant agreement No 101070350 (HPLT).

References

Asaf Amrami and Yoav Goldberg. 2018. Word sense induction with neural bilm and symmetric patterns. *arXiv preprint arXiv:1808.08518*.

Asaf Amrami and Yoav Goldberg. 2019. Towards better substitution-based word sense induction. *arXiv preprint arXiv:1905.12598*.

Nikolay Arefyev, Boris Sheludko, Alexander Podolskiy, and Alexander Panchenko. 2020. Always keep your target in mind: Studying semantics and improving performance of neural lexical substitution. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1242–1255.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90, Montreal, Quebec, Canada. Association for Computational Linguistics.

Terra Blevins and Luke Zettlemoyer. 2020. Moving down the long tail of word sense disambiguation with gloss informed bi-encoders. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1006–1017, Online. Association for Computational Linguistics.

T. Caliński and J Harabasz. 1974. A dendrite method for cluster analysis. *Communications in Statistics*, 3(1):1–27.

Pierluigi Cassotti, Lucia Siciliani, Marco DeGemmis, Giovanni Semeraro, and Pierpaolo Basile. 2023. XL-LEXEME: WiC pretrained model for cross-lingual LEXical sEMantic changE. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1577–1585, Toronto, Canada. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Paul Cook, Jey Han Lau, Diana McCarthy, and Timothy Baldwin. 2014. Novel word-sense identification. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1624–1635, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

V. Dahl. 1909. *Explanatory Dictionary of the Living Great Russian Language ed. by Boduen de Kurtene [Tolkovy slovar zhivogo velikorusskogo yazyka, pod red. I. A. Boduena de Kurtene]*. Helsinki: Kotimaisten kielten keskuksen verkkojulkaisuja 38.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Katrin Erk. 2006. Unknown word sense detection as outlier detection. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 128–135, New York City, USA. Association for Computational Linguistics.

Mariia Fedorova, Timothee Mickus, Niko Tapio Partanen, Janine Siewert, Elena Spaziani, and Andrey Kutuzov. 2024. AXOLOTL'24 shared task on multilingual explainable semantic change modeling. In *Proceedings of the 5th Workshop on Computational Approaches to Historical Language Change*, Bangkok. Association for Computational Linguistics.

Lawrence J. Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification*, 2:193–218.

Denis Kokosinskii and Nikolay Arefyev. 2024. Multilingual substitution-based word sense induction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 11859–11872, Torino, Italy. ELRA and ICCL.

Mikhail Korobov. 2015. Morphological analyzer and generator for russian and ukrainian languages. In *Analysis of Images, Social Networks and Texts*, pages 320–332, Cham. Springer International Publishing.

Andrey Kutuzov and Lidia Pivovarova. 2021. Rushifteval: A shared task on semantic shift detection for russian. In *Computational linguistics and intellectual technologies*, 20, Russian Federation.

Severin Laicher, Sinan Kurtyigit, Dominik Schlechtweg, Jonas Kuhn, and Sabine Schulte im Walde. 2021. Explaining and improving BERT performance on lexical semantic change detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 192–202, Online. Association for Computational Linguistics.

Jey Han Lau, Paul Cook, Diana McCarthy, David Newman, and Timothy Baldwin. 2012. Word sense induction for novel sense detection. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 591–601, Avignon, France. Association for Computational Linguistics.

Jonathan Lautenschlager, Emma Sköldberg, Simon Hengchen, and Dominik Schlechtweg. 2024. Detection of non-recorded word senses in english and swedish. *Preprint*, arXiv:2403.02285.

Xianghe Ma, Michael Strube, and Wei Zhao. 2024. Graph-based clustering for detecting semantic change across time and languages. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1542–1561, St. Julian's, Malta. Association for Computational Linguistics.

Timothee Mickus, Kees Van Deemter, Mathieu Constant, and Denis Paperno. 2022. Semeval-2022 task 1: CODWOE – comparing dictionaries and word embeddings. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1–14, Seattle, United States. Association for Computational Linguistics.

S. A. Mikhaylov and D. M. Shershneva. 2019. Dictionary aggregator vyshka.dictionaries. In *Proceedings of the International Conference on Computational Linguistics and Intellectual Technologies (Dialogue)*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

George A Miller, Martin Chodorow, Shari Landes, Claudia Leacock, and Robert G Thomas. 1994. Using a semantic concordance for sense identification. In *In Proceedings of the workshop on Human Language Technology*, pages 240–243. Association for Computational Linguistics.

Sunny Mitra, Ritwik Mitra, Suman Kalyan Maity, Martin Riedl, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2015. An automatic approach to identify word sense changes in text media across timescales. *Natural Language Engineering*, 21(5):773–798.

Martin Pömsl and Roman Lyapin. 2020. CIRCE at SemEval-2020 task 1: Ensembling context-free and context-dependent word representations. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 180–186, Barcelona (online). International Committee for Computational Linguistics.

Ondřej Pražák, Pavel Přibáň, Stephen Taylor, and Jakub Sido. 2020. UWB at SemEval-2020 task 1: Lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 246–254, Barcelona (online). International Committee for Computational Linguistics.

Maxim Rachinskiy and Nikolay Arefyev. 2021a. GlossReader at SemEval-2021 task 2: Reading definitions improves contextualized word embeddings. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 756–762, Online. Association for Computational Linguistics.

Maxim Rachinskiy and Nikolay Arefyev. 2021b. GlossReader at SemEval-2021 task 2: Reading definitions improves contextualized word embeddings. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 756–762, Online. Association for Computational Linguistics.

Maxim Rachinskiy and Nikolay Arefyev. 2022. GlossReader at LSCDiscovery: Train to select a proper gloss in English – discover lexical semantic change in Spanish. In *Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change*, pages 198–203, Dublin, Ireland. Association for Computational Linguistics.

Digital resource. 1997. *Vanhan kirjasuomen sanakirja [Dictionary of Old Literary Finnish*. Helsinki: Kotimaisten kielten keskuksen verkkojulkaisuja 38.

Dominik Schlechtweg. 2023. *Human and Computational Measurement of Lexical Semantic Change*. Ph.D. thesis, University of Stuttgart, Stuttgart, Germany.

Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2020. SemEval-2020 Task 1: Unsupervised Lexical Semantic Change Detection. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain. Association for Computational Linguistics.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

Frank D. Zamora-Reina, Felipe Bravo-Marquez, and Dominik Schlechtweg. 2022. LSCDiscovery: A shared task on semantic change discovery and detection in Spanish. In *Proceedings of the 3rd International Workshop on Computational Approaches to Historical Language Change*, Dublin, Ireland. Association for Computational Linguistics.

A Ablation study of the NSD model

In this section we provide an ablation study of our NSD model. In order to get insights about the importance of the chosen features, we compare our trained classifiers and several pure similarity measures such as the predicted probability, the dot product and the distance from a usage to the gloss selected by GlossReader FiEnRu (Figure 5). It turns out that the probabilities and dot products are far behind the classifiers, and on the Finnish dev set they even perform no better than a random classifier. The manhattan distance with l1 normalization is a bit worse than the trained classifiers. The extra

Model	dev fi AP		dev ru AP	
	GR	GR FiEnRu	GR	GR FiEnRu
single features				
cosine	0.106	0.110	0.685	0.695
euclid.	0.106	0.110	0.684	0.694
l2/euclid.	0.106	0.110	0.685	0.695
manh.	0.106	0.113	0.685	0.690
l1/manh.	**0.154**	**0.242**	**0.816**	**0.822**
full classifiers				
classifier w/ extra	**0.378**		**0.840**	
classifier w/o extra	0.305		0.833	
best pairs of features w/o extra features				
l1/manh. + euclid.	0.194	**0.284**	0.818	**0.823**
l1/manh. + l2/euclid.	**0.195**	**0.284**	0.818	**0.823**
l1/manh. + manh.	0.192	0.277	**0.819**	**0.823**
best pairs of features w/ extra features				
l1/manh. + #old usages	0.190	**0.291**	0.820	0.827
l1/manh. + #new usages	0.153	0.249	**0.821**	**0.829**
#new usages + #old senses	**0.266**	0.266	0.643	0.643

Table 5: Average precision of novel sense detection models on the dev sets. Except for the block with full classifiers, models use distance-based features either from GlossReader or GlossReader FiEnRu. The best results in each group are in **bold font**. The overall best results are underlined.

features consistently help on the Finnish dev set, but are almost useless on the Russian dev set.

In Table 5 we compare different NSD models using the average precision on the dev sets. To understand which quality can be achieved using the minimal number of features, we evaluate all single distance-based features. Furthermore, we train classifiers on all possible pairs of features, where each pair contains distances only from the same GlossReader. Also we compare classifiers with or without extra features.

We observe that the manhattan distance with l1 normalization, which is the best single feature, works poorly on the Finnish dataset, especially for the embeddings from GlossReader that was not fine-tuned on the Finnish train set. However, on the Russian dev set it closely follows the best classifier. As for the classifiers, we found that including non-distance features is important for Finnish. What is more interesting, when using the original GlossReader model among all pairs of features the best one does not contain embedding-based features at all, only the number of old senses and the number of new usages. This signals that for the Finish dataset GlossReader provides poor embeddings without fine-tuning.

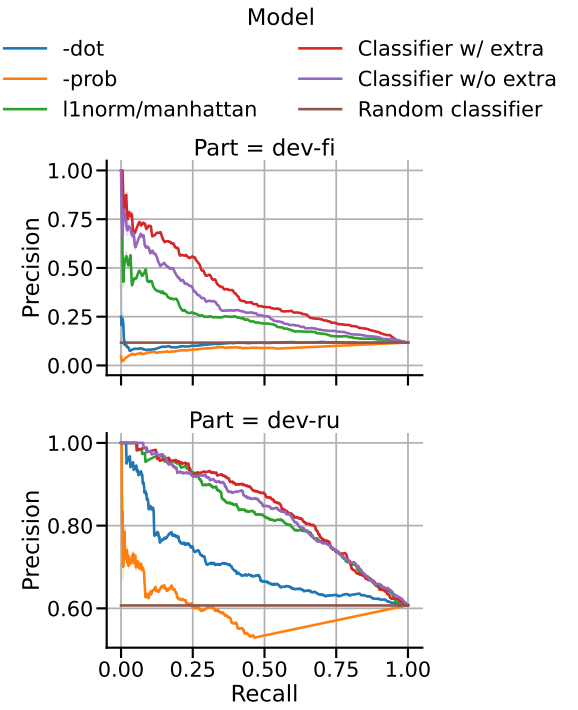

Figure 5: Precision-recall curves of novel sense detection models. Non classifier models are distances between usages and chosen glosses from GlossReader FiEnRu. Classifier w/ extra stands for classifier trained on distance-based and non distance-based features introduced in sub subsection 3.4.3. Classifier w/o extra stands for classifier trained only on distance-based features.