Adapting Semantic Role Labeling to New Genres and Languages

by

Skatje Myers

B.A., University of Minnesota-Morris, 2010

M.S., University of Colorado at Boulder, 2016

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

2023

Committee Members:

Prof. Martha Palmer, Chair

Prof. Mans Hulden

Prof. Katharina Kann

Prof. James Martin

Prof. Vivek Srikumar

Myers, Skatje (Ph.D., Computer Science)

Adapting Semantic Role Labeling to New Genres and Languages

Thesis directed by Prof. Martha Palmer

Semantic role labeling (SRL) is the identification of semantic predicates and their participants within a sentence, which is vital for deeper natural language understanding. State-of-the-art SRL models require annotated text for training, but those annotations don't exist for many languages and domains. The ability to annotate new corpora is hampered by limited time and budget. We explore two different ways of reducing the annotation required to produce SRL systems for new domains or languages: active learning and annotation projection.

Active learning reduces annotation requirements by selecting just the most informative training instances through an iterative process of training and annotation. In this work, we investigate the use of Bayesian Active Learning by Disagreement, ways of tuning it for SRL, and assessing its performance across multiple corpora. We study the choices being made by different selection methods over the course of iterations, examining vocabulary coverage, diversity, predicates selected, and the shifts in confidence. We also explore the impact of various strategies of selecting the initial training data. We investigate a number of potentially influential factors within batches of queries, such as diversity and disagreement scores. In order to reduce the overhead of training time, we additionally compare the effect of increasing the amount of queries being selected on each iteration.

Abstract Meaning Representations (AMRs) are increasingly popular semantic representations of whole sentences. Based on our successful results using active learning to assess the informativeness of annotation instances for SRL, we look into whether the commonalities between these representations can be leveraged to supply targeted annotation for AMR parsing.

Finally, we explore annotation projection of SRL. This approach attempts to create semantic annotations in a target language given parallel translations that have been given SRL annotations through manual or automatic means. We assess the recently developed Russian PropBank and the feasibility of generating the same semantic annotations by projecting from the English PropBank annotation. We use both our own system with English-Russian automatic word alignments and the recent Universal PropBanks 2.0. We examine the types of errors that arise from inconsistencies or gaps in annotations as well as systemic issues arising from the strong English-bias of the projections. This analysis leads us to the development of several filtering techniques that improve the precision of the projections.

Acknowledgements

I first want to thank my advisor, Martha Palmer for her guidance, support, and encouragement. Her mentorship and wisdom have been indispensable. I would also like to thank the rest of my committee – Jim Martin, Mans Hulden, Katharina Kann, and Vivek Srikumar – for their feedback and direction. I also thank the many graduate students, scientists, and faculty members with whom I have collaborated with over the years.

Much of NLP would not be possible without the huge effort that so many have put into creating the corpora that we use. I owe a huge amount of gratitude to Kristin Wright-Bettner, Katie Conger, Julia Bonn, Jeanette Preciado, Tim O'Gorman, Sarah Moeller, Roman Khamov, Adam Pollins, Rebekah Tozier, and so many others, past and present, who have had a hand in the messy process of developing, annotating, and supporting the resources we rely on.

Finally, I would like to give an enormous thanks to my husband, Kyle Hughart, who has been a source of love, support, and reassurance throughout my career and my daughter, Iliana, for her patience and providing me with motivation to do better and work harder every day.

We gratefully acknowledge the support of DARPA KAIROS Program No. FA8750-19-2-1004, DARPA AIDA FA8750-18-2-0016 (RAMFIS), NIH: 5R01LM010090-09 (THYME, Temporal Relation Discovery for Clinical Text), NSF ACI 1443085: DIBBS (Porting Practical NLP and ML Semantics from Biomedicine to the Earth, Ice and Life Sciences), NSF 1764048 RI: Medium: Collaborative Research: Developing a Uniform Meaning Representation for Natural Language Processing. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA, NSF, or the U.S. government.

Contents

Chapter

List of Tables

Table

List of Figures

Figure

Chapter 1

Introduction

The ability to identify the semantic elements of a sentence (*who* did *what* to *whom*, *where*, and *when*) is crucial for machine understanding of natural language and downstream tasks such as information extraction [59], question-answering systems [106], text summarisation [64], and machine translation [79]. The process of automatically identifying and classifying the predicates in a sentence and the arguments that relate to them is called semantic role labeling (SRL). The current state-of-the-art semantic role labeling systems are based on supervised machine learning and rely on large corpora in order to achieve good performance. Large corpora have been created for some languages such as English [103], but such resources are lacking in most other languages. Additionally, those corpora may not translate well to other in-language domains due to sentence structure or domain-specific vocabulary. The creation of annotated corpora requires a significant amount of time and often the hiring of domain experts. It entails establishing new lexical databases defining role categories for verbs, creating annotation guidelines, and annotating large amounts of data. This causes a bottleneck for developing advanced NLP tools for other languages and domains.

In this work, we will be focused on the task of creating new, high-quality corpora for new domains and languages and two ways to make this process more efficient: active learning and annotation projection.

Active learning focuses on choosing only the most useful instances to be annotated, thereby reducing the total annotation requirements to train a supervised model, without sacrificing performance. This is done by iteratively re-training the model and assessing its confidence in its predictions (a proxy for "usefulness") in order to choose additional data for annotation that will have maximal impact on the learning rate. The ideal means by which to choose "useful" instances is still an open question.

In Chapter 3, we investigate the application of Bayesian Active Learning by Disagreement (BALD) as a means of identifying training instances with low model confidence. We evaluate methods of tuning the standard active learning formula to the task of SRL. We will describe our experiments on different methods of estimating model confidence and aggregating multiple predicates into a single score for ranking sentences by informativeness. We then present our research on the impact of selecting entire sentences against selecting individual predicates and its varying effect on multiple corpora.

Furthermore, we seek to better understand the decisions made by active learning for SRL and identify possible avenues of further improvement. To this end, we analyse the batches of selected sentences over the course of the process across multiple domains. While performance plateauing is a straightforward indication of the limited utility of continuing to apply active learning, we also investigate whether the disagreement scores provided by BALD can shed light on how much further improvement can be had.

Because the active learning process itself incurs the cost of re-training a model and disruption to annotation workflow, we examine the impact of selecting differing sizes of batches to annotate on each iteration. Building on previous work, we also test methods of choosing a starting seed set with greater diversity and atypical sentences in order to increase efficiency.

Abstract Meaning Representations (AMRs) are graph structures representing the meaning of a sentence that incorporate PropBank frames along with capturing additional semantic details, such as named entities, noun modifiers, discourse connectives, and intra-sentential coreference. While AMRs have been increasingly utilised for many downstream tasks (e.g., machine translation [52], text summarisation [62], and knowledge base question answering [41]) most of the existing training data is in the general news domain. Applying BALD to these structures is less straightforward than for a sequence-labeling task like SRL. Since there is significant overlap between the two semantic representations, we investigate in Chapter 4 whether active learning for SRL can also supply informative instances to target for AMR annotation and make the development of parsing models more efficient.

In Chapter 5, we will examine annotation projection for SRL. Annotation projection leverages parallel corpora, where the source language has automatic or manual semantic roles, to create annotated corpora in the other target languages. Approaches to this task typically use unsupervised word alignments, filtering heuristics to improve precision, and bootstrapping an SRL model to improve recall.

We present our initial results and analysis on projecting English annotations into manually translated sentences that have been annotated by Russian PropBank, developing language-specific filtering techniques to improve the results. Through this analysis, we identified issues for cleanup and expansion of Russian SRL data. We also provide additional context on differences between Russian and English semantics.

The recent Universal Propbanks 2.0 [40] uses bootstrapping techniques to improve recall and better handle cases where the parallel sentences are imprecise translations. This framework has been used to create SRL corpora in 23 languages. We evaluate this system's projections against Russian PropBank and examine the errors for systemic issues that can be incorporated into the existing filtering methods and then evaluate these improvements.

Finally, we will summarise our conclusions from these experiments on expanding semantic resources to new domains and languages in Chapter 6.

Chapter 2

Background

In this chapter, we first describe semantic representations and the task of semantic role labeling. First we will describe two styles of semantic annotations that are relevant to our work and some of the relevant corpora. Following this, we will discuss active learning broadly as well as the prior literature that used this technique for semantic role labeling in particular. Next we will describe another form of semantic representation, Abstract Meaning Representations, and how they relate to SRL. Finally, we will survey previous work on transferring semantic annotations from English into other languages.

2.1 Semantic Role Labeling

Semantic role labeling (SRL) is a process of assigning labels to entities in a sentence, indicating the semantic relations between them. Observe the following two sentences:

- John broke the window.
- The window was broken by John.

Although the syntax of these sentences differs, the underlying event being described is the same. In both cases, *John* is the one who performed the action of *breaking*, and the thing that was broken was the *window*. The task can be formulated as 1) identification of predicates [‘break’] and assignment to a frame denoting sense [break.01: ‘cause to not be whole’], 2) identification of participants [‘John’, ‘window’] and assignment of role types [‘agent’, ‘patient’, respectively].

This shallow semantic information from sentences can be fed into systems for many downstream NLP tasks, such as information extraction [59][107], question-answering systems [106][24], natural language inference [109], text summarisation [64], and machine translation [60][79], that benefit from using semantic features.

Automatic semantic role labeling is evaluated with respect to precision (the percent of automatic predictions that are correct), recall (the percent of correct arguments that were predicted by the system), and the F-score (the harmonic mean of precision and recall).

2.1.1 Semantic Lexicons

There have been several proposed ways of representing semantic roles and predicates. Proposition Bank and FrameNet provide differing ontologies of predicate frames and semantic roles.

The FrameNet project [6] aims to document semantic frames that group together semantically related verbs, providing a prototypical representation of the situation. For example, in a sentence containing "argue", "banter", or "debate", the frame "Conversation" is evoked. For a sentence such as this, the semantic roles to be used are determined by this frame, and include "Protagonist1", "Protagonist2", and "Topi". As will be later discussed in section 2.4, some of the prior work done on cross-lingual projection focused on projecting this style of annotation.

Proposition Bank (PropBank) [74][76] takes a more verb-oriented approach than FrameNet, while seeking to represent semantics in a more generalisable way. Contrary to FrameNet, the list of permissable roles is defined by the sense of each verb. Words like "argue" and "banter" aren't grouped together, but instead the presence of "argue" invokes a specific roleset that determines the available arguments. Rather than arguments having specific names such as "Protagonist1", they are given generalised numbered labels, ARG0 through ARG5. Typically an ARG0 is similar to a Proto-agent (per Dowty, 1991 [23]), and is the agent or experiencer, while ARG1 is typically the patient or theme of the predicate, similarly to a Proto-patient. By generalising the arguments in this way, automatic semantic role labelers can produce useful information even if they misidentify the frame. Additionally, there are modifier arguments to incorporate other semantically relevant

Roleset id: give.01	
transfer	
Arg0	giver
Arg1	thing given
Arg2	entity given to

Table 2.1: PropBank roleset for *give.01.*

information such as location (ARGM-LOC) and direction (ARGM-DIR). PropBank is the type of semantic annotation that our work on cross-lingual annotation projection and active learning is concentrated on.

The following is an example of the arguments related to the predicate “give” according to the roleset in Table 2.1:

[$_{\text{ARG0}}$ She] had [$_{\text{give.01}}$ given] [$_{\text{ARG1}}$ the answers] [$_{\text{ARG2}}$ to two low-ability geography classes].

Sentences may contain several predicates and each predicate has its own arguments. Predicates may consist of only verbs in some annotation schemas, but may also include nominalisations and predicative adjectives.

Annotations may be span-based and placed in alignment with a constituent parse, or placed on head words for compatibility with dependency parses.

2.1.2 Corpora

Many large corpora have been annotated in English, such as Ontonotes [103]. Although Ontonotes has since been retrofitted to unify different parts of speech into the same rolesets based on sense and given expanded nominalisations, light verb constructions, and other multi-word expressions [69], an earlier version of it was released as the dataset for the CoNLL-2012 shared task. This dataset is still frequently used as an evaluation corpus for experimental SRL techniques. Our active learning experiments in Chapter 3 are performed with the latest version of Ontonotes, version 5.0.

Additionally, there are many domain-specific SRL corpora, such as clinical records [3] and the geosciences [25]. These domain-specific annotations are necessary because the vocabulary and sentence structure may differ too much for models trained on more general text to perform well [77][3].

Semantic corpora have been manually developed in other languages using PropBank-style annotations, such as Hindi and Urdu [9]; Arabic [73]; Chinese [104]; and Korean [91].

The Russian FrameBank [58] is a project to develop frames designed for Russian and annotate examples of those frames from the Russian National Corpus. Their annotation scheme uses 96 distinct semantic roles, such as Result or Beneficiary, similarly to FrameNet (which has over 1k such roles), and organised in a hierarchical graph. They constructed frames for approximately 4000 target verbs, adjectives, and nouns, and annotated over 50,000 examples of these frames.

The Universal PropBanks (UPB) [1] are a collection of semi-automatically generated corpora in multiple languages. The first release consisted of 7 languages [2], and has since expanded to 23 languages in the Universal PropBanks 2.0 release [40]. The earlier version created these datasets from parallel governmental text using word alignments and then were hand-curated, while the expansion into additional languages was facilitated by using parallel sentences from OPUS [97]. We will discuss the methodology of its creation further in Chapter 5.

X-SRL [19] is another automatically developed parallel SRL corpus. The authors projected the English CoNLL-09 dataset to automatic German, French, and Spanish translations. The test dataset was human-validated. As with UPB, their projection included only verbal predicates.

The Low Resource Languages for Emergent Incidents (LORELEI) project [2] sought to explore techniques to rapidly develop natural language processing technologies for low-resource languages. The dataset released as part of this project consists of parallel corpora for 23 low resource languages. The “core” data consists of approximately 550,000 tokens (in English) of newswire, phrasebook, social-network, weblog, discussion forum, and elicited text. A portion of the newswire, phrasebook,

[1] https://github.com/System-T/UniversalPropositions

[2] https://www.darpa.mil/program/low-resource-languages-for-emergent-incidents

and elicited text were manually annotated with SRL.

A subset of this corpus, consisting of *newswire* and *phrasebook* sentences in Russian, was annotated with PropBank-style semantic roles [63]. The Russian PropBank is in active development. The experiments we will present on annotation projection in Chapter 5 utilise two versions of this dataset, which we will refer to as *Russian PropBank 2020* and *Russian PropBank 2023*. These will be described in more detail in Sections 5.2.1 and 5.3.

2.2 Active Learning

Many state-of-the-art NLP systems rely on supervised machine learning models that are trained on large amounts of hand-annotated data. Creating a sufficiently large training corpus through human annotation requires significant cost in terms of both time and money.

One way of reducing this burden is by selecting only the most informative instances to annotate. Active learning (AL) allows the learner to choose the data it would like the annotator to label next, selecting what is thought to be the most informative instances to learn from. Just as a person who can ask questions will learn faster than if they are simply lectured, a model will too. The literature shows active learning can produce a model with equal, or sometimes better, performance with a fraction of the data needed by a model trained on randomly-selected data. [83]

Active learning begins with the selection of a model suitable for the task to be learned, a small pool of labeled training data (also referred to as a *seed set*) for the model to initially be trained on, and a large amount of unlabeled data. AL is an iterative process where the model is trained on the labeled data and then through some *query selection strategy*, an instance or instances are chosen from the available unlabeled data to request an oracle (in practice, a human) to provide a label for. Most typically, they're chosen after the model attempts to predict labels for the unlabeled data and provides feedback about what instances may be the most informative. The newly annotated data from the oracle is then added to the pool of labeled data that will be used to train the model on the next iteration. This iteration continues until some stopping criteria are met, such as the model becoming above a certain threshold of confidence about the remaining unlabeled data, or simply until funds or time are exhausted.

```
$L$ = labeled training data;
$U$ = unlabeled data;
$M$ = a model;
while stopping criteria not met do
    Train $M$ on $L$;
    $U'$ = select instance(s) from $U$;
    $L'$ = annotate $U'$;
    $L = L \cup L'$;
    $U = U \setminus U'$;
end
```

Algorithm 1: Generalized active learning

2.2.1 Active Learning Applied to SRL

While the usefulness of active learning has been demonstrated for numerous NLP tasks [108], including named entity recognition [84], word sense disambiguation [110], sentiment classification [53], part of speech tagging [15], and natural language inference [89], research on its use for semantic role labeling is still in its early stages.

Since probabilities from off-the-shelf NN models may sometimes be inaccessible, Wang et al. [101] proposed working around this by designing an additional neural model to learn a strategy of selecting queries. Given an SRL model's predictions, this query model classifies instances as requiring human annotation or not. Their approach was a hybrid of active learning and self-training. The self-training is enacted by accepting the SRL model's predicted labels into the training pool for future iterations when the sentence was determined not to require human annotation. This approach requires 31.5% less annotated data to achieve comparable performance as training on the entirety of the CoNLL-2009 dataset.

Koshorek et al. [48] compared data selection policies while simulating active learning for question-answer driven SRL (QA-SRL). QA-SRL is a form of representing the meaning of a sentence using question-answer pairs. Rather than annotating spans of text with argument names, such as PropBank's ARG0, annotators enumerate a list of questions relating to the actions in a sentence, such as *who* is performing an action and *when* is it happening, along with the corresponding answers from the original text. This representation provides similar coverage to PropBank, but can also represent implicit arguments that aren't directly represented by the syntax.

The process of identifying spans that are arguments of a predicate and the generation of questions based on the arguments were treated as independent tasks. To provide an approximate upper bound on the learning curve, they simulated active learning on the dataset, splitting the unlabeled candidates into K subsets, each comprised of L examples, and selecting the subset that improved the model the most on the evaluation data. Against this oracle policy, they compared the following selection strategies, sampling K random subsets to choose from: selecting a random subset, selecting the subset with the highest average token count among sentences, and selecting the subset that has the maximal average entropy over the model's predictions.

In their experiments detecting argument spans for predicates, they used five subsets of one example each. The oracle policy reached a 9% improvement above random selection, with a diminishing effect as the number of training examples increases. The selection of longer sentences outperformed the oracle policy. The uncertainty strategy performed worse than random selection for argument span detection, and was not tested for question generation. Selecting the sentences with high token counts tended to improve the F-score for argument span detection by 1-3% given an equal number of training instances (and attaining 60% on the full dataset), while being largely comparable to random selection for question generation.

Active learning for SRL has also been applied in combination with multi-task learning [35], using a subset of PropBank roles along with a new "Greet" role. The authors compared single- and multi-task SRL, both with and without active learning. Under multi-task learning the model jointly learns to identify semantic roles, as well as to classify tokens as entities such as "Person" or "Location". They introduced a set of semantic roles that accommodate conversational language and annotated a small corpus of Indonesian chatbot data to provide training and testing data. By selecting sentences using model uncertainty in the single-task context, F-score was improved by less than 1% compared to randomly selecting the data.

In order to improve computational efficiency in selecting new labeled instances, Houlsby et al. [33] proposed Bayesian Active Learning by Disagreement (BALD). Now that neural networks dominate supervised learning, methods based on this have been shown to be effective as an improvement over using the output layer of the NN to determine the most beneficial new instances [85][88]. We will provide further background on the application of BALD to SRL in particular in Chapter 3.

2.2.2 Seed Selection

Tomanek et al. [98] showed that a seed set that is biased towards rare class instances will help avoid the "missed class effect", a form of the "missed cluster effect" [82], where entire clusters or classes of data can escape selection due to insufficient exploration of the data space. In their experiment using AL for NER, these instances were selected with the prior knowledge of the labeled class of the instances and the class distribution. Since the goal of active learning is to avoid the need for large amounts of annotation, utilizing this finding in practice relies on obtaining informative rare class instances through an unsupervised or semi-supervised approach.

Dligach and Palmer [20] trained an unsupervised language model (LM) on the datasets they were performing active learning for word sense disambiguation (WSD) for, targeting verbs specifically. The verb itself as well as the surrounding words from both sides within a three-word window were provided as features to the LM. Once the model is trained, one can calculate the perplexity of a given instance – or in other words, a measurement of how well the model can predict that instance. In this way, instances can be ranked from least probable to most probable according to what the model has seen. The authors showed that rare verb sense classes are more concentrated among those instances in the half of the rankings with lower probability. Using the least probable instances to seed the active learning of a WSD model provided an improvement over choosing those initial sentences randomly. Commonalities between WSD and the SRL task suggests that this seeding technique may also be beneficial to our work.

2.3 Abstract Meaning Representations

Abstract meaning representations are rooted, labeled, directed, acyclic graphs that aim to represent the semantics of a whole sentence [7]. Nodes are comprised of concepts – such as words from the sentence like “they”, named entities like “country”, or roleset IDs – and are given variable names for reference. These variables allow for intra-sentential coreference. AMRs primarily use PropBank frames to capture predications in the sentence. For example:

```
They live in the south of France.
(l / live-01
   :ARG0 (t / they)
   :location (s / south
         :part-of (c / country :wiki "France"
               :name (n / name :op1 "France"))))
```

In this case, “they” is identified as an ARG0 according to the *live.01* PropBank frame. While PropBank annotation would label ARGMs for any non-core arguments to the predicate, AMRs have their own set of relation types that replace these (such as :location instead of ARGM-LOC, :purpose instead of ARGM-PRP, :polarity instead of ARGM-NEG) as well as several additional ones. AMRs are able to represent additional semantic relations such as :part-of, as well as to separate concepts into their constituent parts, such as using :day, :month, and :year to connect nodes representing the sub-components of a date-entity. AMR extends the original LDC Named Entity tag set substantially and adds Wikipedia links (as in the above example) for well-known Persons, Locations and Organizations.

Abstract meaning representations have been valuable for many NLP tasks, such as machine translation [93], text summarisation [57], and knowledge base question answering [41].

Semantic role labeling can be considered a subtask of AMR parsing and SRL has been successfully used as an intermediary task to improve performance on AMR parsing [16].

AMR parsing performance is evaluated by the SMATCH score [14], which measures the degree of overlap between the two semantic structures.

2.3.1 Corpora

Many English AMR corpora have been created. The largest corpus is the AMR Annotation Release 3.0 [44], consisting of newswire, broadcast conversation, discussion forums, weblogs, Aesop's fables, and Wikipedia text from multiple genres. Other available corpora include The Little Prince [3] and a corpus in the biomedical domain [4] .

Additionally, AMRs have been developed in other languages, such as Chinese [51], Spanish [102], Brazilian Portuguese [90], Korean [17], Vietnamese [55], and Turkish [70].

2.4 Annotation Projection

Annotation projection is the method of transferring annotations from one language to another using parallel text. This technique has been utilised for numerous other NLP tasks, such as part-of-speech tagging [105], named entity recognition [27], syntactic dependencies [34][96], and abstract meaning representations [86]. Prior work on SRL projection has been successfully used for projecting annotation from English to French [100], German [72], Turkish [1], and a variety of other languages [38]. Some of the early work in this area, such as Padó and Lapata [72], focused on transferring FrameNet to other languages, but more recent work has focused on PropBank instead due to availability of high-performance SRL models, and its more generalised role types that make it suitable for use in other languages [65].

[3] https://amr.isi.edu/download/amr-bank-struct-v3.0.txt

[4] https://amr.isi.edu/download/2018-01-25/amr-release-bio-v3.0.txt

Annotation projection relies on unsupervised word alignments to provide the correspondences between parallel sentences. In previous work, these alignments have been provided by models such as GIZA++ [68] (used by Van der Plas et al. [100], Padó and Lapata [72], Aminian et al. [5]), Berkeley Aligner [54] (used by Abkik et al. [2]), fast_align [26] (used by Fei et al. [29]), and SimAlign [36] (used by Jindal et al. [40]). The alignment models provide mappings for each word in a source sentence to zero or more words in a target sentence. Typically, one performs the alignment from one language to another, then vice versa, and takes only the intersection of the alignments in order to reduce errors.

The most straightforward way of using these alignments is Direct Semantic Transfer (DST), which has since become a common baseline for newer projection approaches. This simple method, described by Van der Plas et al. [100], transfers a predicate or argument label from a source-language word to a target-language word if the alignment model generated an alignment between the two. We show an example of this in Figure 2.1.

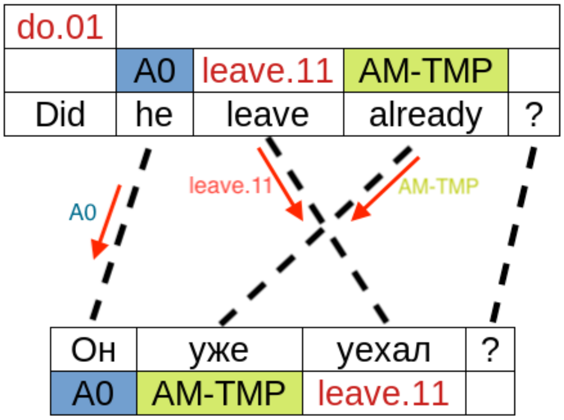


Figure 2.1: Direct Semantic Transfer of PropBank annotations from an English sentence to a Russian sentence, along word alignments (shown as dashed lines).

The advent of robust multilingual embeddings has provided a new avenue for determining these mappings between parallel sentences by enabling a comparison of similarity within a shared representation space. Daza and Frank [19] projected annotation labels from English into French, German, and Spanish using cosine similarity of mBERT to determine alignments between words and generated the parallel X-SRL corpus. Rather than using the intersection of source-to-target and target-to-source word alignments, they used only the source word-pieces to target word-pieces, finding this approach to significantly improve recall. Although it was detrimental to precision, the F-score on the whole improved by 1.5-9.4% compared to using the alignment intersections, depending on language.

There has also been related work on using alignments for the task of projecting Abstract Meaning Representations. This task overlaps significantly with SRL, as AMRs use the same semantic frames. Sheth et al. [86] similarly used embeddings – XLM-R [18] in this case – to provide word alignments for AMRs using cosine similarity.

Since identification of the correct alignments between parallel sentences is itself a challenging task, errors in this step can negatively impact the precision of the projections. In order to improve performance on this, previous work often utilises filtering heuristics and methods to either adjust the alignments or weed out spurious correspondences. We will provide more in-depth background on the error analysis and filtering techniques used by related works in Chapter 5.

After the initial annotations have been projected, training an SRL model on them can provide further improvements and increase recall. The authors of one of the early works on using Direct Semantic Transfer for cross-lingual SRL, Van der Plas et al. [100], trained a joint syntax-semantic model on the word-alignment projections from English to French in order to re-label the data, overwriting the original projections. After this training, the model's performance improved, resulting in labeled predicate and labeled argument F-scores only 4% and 9% below the upper bound set by inter-annotator agreement, respectively.

2.4.1 Bootstrapping

Bootstrapping, or self-training, is a technique where one uses the model's own predictions to improve performance. After training on an initial set of labeled data, the model is used to either predict additional instances to add to the training pool or, in the case where the training data is not manually annotated, to re-label its own training data. Sometimes the choice of which of the model's predictions should be included in the training data is filtered in some manner to only include high quality predictions. In the case of cross-lingual SRL, bootstrapping can provide a much-needed boost to recall caused by inaccurate alignments.

In the process of semi-automatically constructing a multilingual corpus with unified semantic roles for 7 languages, called the Universal PropBanks, Akbik et al. [2] used an iterative bootstrapping method. Initially, they projected annotations from English into each language using DST of automatic SRL labels using word alignments and applying a series of filters, which will be described in more detail in Section 5.1.

The now-labeled target sentences were used to initialise a bootstrapping process, where the model was successively trained on the target data, though limiting it only to target data above a certain threshold of "completeness", where a sufficient number of dependents of the verbs were given SRL labels. They compared using the model to *overwrite* the data, as done by Van der Plas et al. [100], versus *supplementing* the training data. In the latter method, instead of overwriting the projected labels, the model's predictions were used to only add labels for words without projected labels. They found that the predictions tended to lower precision, while increasing recall, so supplementing proved more beneficial than overwriting.

For releasing datasets and evaluation on them, they only chose sentences to which the SRL model had assigned labels for all verbs and their dependents (which reduces the final generated PropBanks to 3%-19% of the size of the original data, depending on language). Manual evaluation on 100 sentences in each target language found extremely high performance for all languages other than Hindi, with precision and recall for predicates being over 95% and 88% respectively and arguments being over 85% and 66% respectively, when considering partial matches.

Universal PropBanks 2.0 [38] is able to achieve additional performance gains through several upgrades over their previous system. They update to using more recent SRL and word alignment models and add several improvements to their bootstrapping approach. They train the bootstrapped SRL model not only on the target language labels, but also gold English SRL, using multilingual embeddings. They additionally jointly train the model on both span-based and dependency-based labels.

Aminian et al. [4] found the bootstrapping method that only fills in missing SRL decisions to slightly underperform overwriting the sentence with the new predictions upon each iteration when projecting from English to German using the CoNLL-2009 shared task dataset. Along with testing those two bootstrapping methods, the authors proposed a cost-sensitive way of updating the model, weighting the penalty for mislabeling an instance with various cost functions. They tested three such measures: 1) As with previous work, the authors consider the *completeness* of the annotations of a sentence (i.e. how many verbs and dependents were given a label) indicative of quality of projection and the likeliness that they contain translation shifts that render it incompatible with the source semantic frames, 2) whether the dependency label of a target word matches that of the aligned source word, and 3) both combined. The difference between these three cost functions was minimal (<1% F1), but the bootstrapping with a combined cost function resulted in an F-score increase of 1.4% over relabeling bootstrapping without a cost function and 3.5% over using a model trained only on the original projections.

2.4.2 Data Requirements

The majority of prior work on cross-lingual SRL relies on access to parallel data, with the source side labeled with semantic annotations. The most common approach [2][99] for procuring this type of data is to obtain automatic labels on the English side of an existing parallel corpus with manual translations, such as Europarl [45] or the UN Parallel Corpus [111], using one of the many available SRL systems. However, automatic systems will always have the potential to mislabel the data.

Alternatively, given an existing corpus with manual semantic annotations, we can manually or automatically translate it into our target language, though the automatic approach can introduce errors through poor translation quality. This translation-based approach has previously been used for cross-lingual syntactic parsing [95], as well as for SRL more recently [29].

Since both of these methods can suffer from noise at the outset either due to automatic

SRL or automatic translations, Cai and Lapata [13] de-couple these two aspects of cross-lingual SRL, and do not require the parallel data to be the same as the data with SRL labels. Instead, their model simultaneously trains on batches of the annotated source language data in a supervised fashion, while also training on batches of the unannotated parallel data using the the model's own predictions for both source- and target-language sentences.

2.5 Summary

In Section 2.1, we have provided the motivation for and explanation of semantic role labeling. We have given overviews of the commonly used semantic representations and many of the datasets commonly used for training and evaluation.

In Section 2.2, we discussed active learning and its usefulness in reducing annotation requirements for supervised models and how this has previously been applied to SRL. We will further delve into the use of Bayesian Active Learning by Disagreement in particular in Chapter 3.

In Section 2.3, we have offered an overview of abstract meaning representations and their relation to PropBank.

In Section 2.4, we review annotation projection as a means of developing semantic resources for new languages. We will provide additional background on the filtering techniques used by prior work in Chapter 5.

In the next three chapters, we will present our work on applying active learning to SRL across multiple datasets, investigating the cross-task application of SRL AL to developing AMR corpora, and developing, evaluating, and improving annotation projection using linguistically motivated error analysis.

Chapter 3

Active Learning for SRL

One of our goals is to further expand research on active learning for the task of semantic role labeling. In Section 3.1, we will provide background on Bayesian Active Learning by Disagreement and prior literature applying it to SRL.

In Section 3.2, we will describe the SRL model and experimental framework that we utilise throughout this chapter.

Motivated by the success of Bayesian Active Learning by Disagreement on previous work, we explore its application to SRL and ways of tuning it for this task. In Section 3.3, we investigate ways of aggregating scoring and assessing its performance compared to conventional model output probabilities with respect to annotation workflow [66].

In Section 3.4, we investigate the impact of selecting and training on individual predicate-arguments structures. These experiments produce varied results across the four datasets tested. In Section 3.4.4, we examine some of the differences between the corpora. In Section 3.4.3, we study the composition of the selections over time with an eye towards vocabulary coverage and sentence and predicate diversity.

While smaller batches of queries per iteration allow for better utilisation of active learning's ability to provide updated assessments of the informativeness of candidate instances, this repeated training requires additional time and computational resources. In Section 3.5, we investigate the ideal number of queries to use for each iteration across varied datasets.

Because active learning's success is largely due to its ability to hone decision boundaries, in

Section 3.6 we explore methods of addressing the potential issue of failing to sufficiently explore the data space by enforcing diversity within the initial training seed.

3.1 Bayesian Active Learning by Disagreement

Modern SRL systems utilise deep learning, which poses a challenge to assessing the model's certainty in its predictions. The predictive probabilities in the output layer cannot be reliably interpreted as a measure of model certainty. Gal and Ghahramani [30] proposed using dropout as a Bayesian approximation for model certainty, estimating it using the variation in multiple forward passes.

This dropout principle was tested on numerous NLP tasks by Siddhant and Lipton [88], including SRL. For their SRL experiments, they used a neural SRL model based on the He et al. [32] model, with modifications to the decoding method (instead using a CRF decoder) and increasing the dropout rate from 0.2 to 0.25.

In comparison to the baseline of random selection, they tested the classic uncertainty measure of using the output probabilities of the model, normalised for sentence length, with two Bayesian Active Learning by Disagreement methods for selecting additional instances: Monte Carlo Dropout Disagreement (DO-BALD) and Bayes-by-Backprop (BB-BALD). The BB-BALD method provides uncertainty estimation by drawing Monte Carlo samples from a Bayes-by-Backprop neural network [10]. The DO-BALD method applies dropout during multiple predictions of instances in the unlabeled pool and selects instances based on how many of those predictions disagree on the most common label of the entire sequence. The authors treat agreement between predictions as all-or-nothing, rather than allowing partial agreement based on arguments or predicates. They calculate disagreement between 100 forward passes per sentence. In our work [66][67], we explore this technique with additional tuning for SRL, which will be described in Chapter 3.

They tested their methods on both the CoNLL-2005 and CoNLL-2012 datasets, which use PropBank annotation. While the Bayesian methods were similar to the standard uncertainty selection method in the case of SRL, these methods resulted in approximately 2-3% increase for F-score compared to random selection when training on the same number of tokens. These results were much more modest than results for other tasks such as NER.

For the two datasets, the authors found that they could obtain the same performance as training on the entire dataset by randomly selecting only approximately 50% of it. Similar success of the random selection baseline was reported on other datasets for other tasks in their work, as well as in other active learning studies [84].

3.2 Experimental Framework

3.2.1 Model

We used AllenNLP's [31] implementation of a state-of-the-art BERT-based model [87]. Our training procedure for this model used 25 epochs or stopped early with a patience of 5. Trained under the same experimental configuration on the full training subsets, this model achieves an F-score of 83.82 and 83.48 on the *OntoNotes* and *THYME* datasets respectively.

After training on the initial seed dataset, each iteration of active learning selected a given number of sentences or predicates and re-trained from scratch. In the case of the whole-document baseline that will be tested in Section 3.3, for the creation of each batch, we selected random documents until the number of sentences selected met or exceeded 100.

3.2.2 Datasets

In this chapter, we will provide a demonstration of active learning for SRL across a variety of domains and sublanguages [42]. Some knowledge domains exhibit narrow lexical, syntactic, and semantic structures that distinguish them from more general-purpose domains. This can dramatically lower performance when testing with a model trained on more general text [3]. Special techniques that take these domain specific-structures into account are needed for adapting NLP tools to these domains, as illustrated below.

THYME Colon is 522k tokens, comprised of unstructured clinical notes relating to treatment of colon cancer [3]. This corpus contains specialised medical vocabulary for a narrow domain and a large number of formulaic sentences, such as the following example:

> Pathology demonstrated a tubular adenoma with moderate dysplasia.

This contains medical terminology (tubular adenoma, dysplasia) as well as a non-standard use of *demonstrate*, which includes the shortening of *The pathology report* to simply *pathology.* This particular framing re-occurs frequently in *THYME Colon*, sometimes with *show* or *reveal* instead, and occasionally including the word *report* as in *pathology report.*

We also used two distinct geoscience domains from the ClearEarth project [25]. *Earthquakes* consists of 41k tokens of text from Wikipedia and education texts, and a glossary. This text includes specialised scientific language relating to earthquakes and plate tectonics, but also discussion of the history of the field at a high school reading level and content related to disasters. An example of this type of data is the following:

> The ways that plates interact depend on their relative motion and whether oceanic or continental crust is at the edge of the lithospheric plate.

Ecology consists of 83k tokens of text from Wikipedia, educational websites, an ecology glossary, and Encyclopedia of Life. The scientific content covers genetics, evolution, reproduction, and food chains. For examples:

> Anguis fragilis is an example of ovo-viviparity.

> Alternatively, transcription factors can bind enzymes that modify the histones at the promoter.

OntoNotes 5.0 [103] spans multiple genres, largely consisting of news sources, but also including telephone conversations, text from the New Testament, weblogs, and Usenet. This popular corpus serves as a broad purpose corpus for us, as opposed to the other more specialised domains.

We use a version of *OntoNotes* that does not include files that had no manual PropBank annotation performed. There still exist sentences within this version of the data that had only partial annotation, but we consider this to have a relatively small impact on performance.

Evaluation was performed on the standard test subset for each respective corpus.

3.3 Tuning Bayesian Active Learning by Disagreement for SRL

Traditionally, AL practitioners use the model's probability distributions for the annotation candidates to quantify how informative a new training instance would be for the model. However, state-of-the-art SRL systems rely on deep learning, whose predictive probabilities are not a reliable metric of uncertainty [30]. As discussed previously in section 3.1, Bayesian Active Learning by Disagreement [33] is a strategy of measuring model uncertainty by calculating the rate of disagreement of multiple Monte Carlo draws from a stochastic model.

Semantic role labeling for a single sentence is a complicated structural prediction, involving multiple predicates and varying spans. This complexity makes identifying the training examples with maximal impact more challenging. In this section, we compare two ways of aggregating confidence scores for individual predicates into a unified score to assess the usefulness of selecting a sentence for active learning. We test these strategies with two active learning approaches to calculating certainty for a predicate instance: the model's output probabilities and a granular DO-BALD selection method. Additionally, we compare the benefits of these AL approaches with three baselines: random sentence selection, random document selection, and selecting sentences with the most predicates.

3.3.1 Data

We used two corpora for these experiments, as previously described in detail in Section 3.2.2: The English section of *OntoNotes* (version 5.0) [103] with the latest frame updates [69] and the *THYME Colon* corpus [3].

We simulated active learning on the training subset of each corpus, dividing it into an initial seed set and a set of sentences to select from. The initial seed sets for sentence-based experiments were 200 randomly chosen sentences. For the whole-document baseline, the seed set is comprised either of documents from multiple genres, totalling 200 sentences, in the case of *OntoNotes*; or a single patient (consisting of two clinical notes and one pathology report, totalling 195 sentences) in the case of the *THYME Colon* corpus.

In both cases, we utilised validation data to determine early stopping. Due to the excessive computational time required to predict the standard validation sets for these corpora for every epoch for every iteration, as well as the fact that a real-world scenario would be unlikely to have such a disproportionally large validation set to perform active learning, we selected a subset of the validation data for use. In the experiments involving selecting individual sentences, we used the same randomly chosen 250 sentences. In the case of the baselines of choosing random documents, we used validation datasets approximating 250 sentences, comprised of whole documents.

3.3.2 Selection Methods

3.3.2.1 DO-BALD

The model output of neural networks give a poor estimate of confidence, due to their nonlinearity and tendency to overfit and be overconfident in their predictions [30][21].

Using Monte Carlo dropout as a Bayesian approximation of uncertainty, as proposed by Gal and Ghahramani [30], we applied a dropout rate of 10% during the prediction stage. We employ the Bayesian Active Learning by Disagreement approach by predicting each candidate sentence multiple times to select sentences based on how often those predictions agree with each other.

The number of predictions used correspondingly increases the time required to select data upon each iteration. Gal and Ghahramani [30] used between 1000 and 10 forward passes in their experiments and Siddhant and Lipton [88] used 100 per sentence when applying DO-BALD to SRL. An ideal solution would minimise this variable for efficiency with as little loss as possible in the benefit gained by sampling the distribution. In our experiments, we chose to perform 5 predictions per predicate. Due to sentences containing multiple predicates, this typically results in 10-15 predictions per sentence.

From these predictions, agreement was calculated based on entire argument spans. For each predicate in the sentence, we considered the percent of predictions for each argument type that agreed with the most frequent span choice for that type. Referring to the example in Table 3.1, the most frequently chosen span for ARG0 was “John Smith”, although two of the predictions chose only the partial match of “John”. In this case, since two out of the five disagree with the most common prediction, the argument ARG0 has a disagreement rate of 0.4. The rate of disagreement was calculated for each argument type present in the set of predictions and then averaged to summarise the consensus for the entire predicate-argument structure.

Prediction 1	[$_{\text{ARG0}}$ John Smith] [$_{\text{Pred}}$ bought] [$_{\text{ARG1}}$ apples].
Prediction 2	[$_{\text{ARG0}}$ John] Smith [$_{\text{Pred}}$ bought] [$_{\text{ARG1}}$ apples].
Prediction 3	[$_{\text{ARG0}}$ John Smith] [$_{\text{Pred}}$ bought] [$_{\text{ARG1}}$ apples].
Prediction 4	[$_{\text{ARG0}}$ John Smith] [$_{\text{Pred}}$ bought] [$_{\text{ARG1}}$ apples].
Prediction 5	[$_{\text{ARG0}}$ John] Smith [$_{\text{Pred}}$ bought] [$_{\text{ARG1}}$ apples].

Table 3.1: An example of varying argument predictions for a predicate, *bought*, by multiple forward-passes with dropout.

By examining the forward-pass predictions predicate-by-predicate and argument-by-argument to determine agreement, our approach is more granular than Siddhant and Lipton’s [88] method of determining disagreement from the mode of the entirety of the sentence’s labels. Our strategy allows for partial credit when the predictions are in agreement about particular arguments.

3.3.2.2 Combining Predicate Scores

Since sentences often contain multiple predicates, we must aggregate the scores into a single measure in order to rank sentences by their potential informativeness. We propose two such ways of combining the predicate scores, which we applied to both the *model output* and DO-BALD methods of calculating certainty of a single predicate-argument structure:

- **Average of Predicates (AP)**: The score for all predicate-argument structures in a sentence is averaged. This provides a balance between the predicates in the sentence, but high confidence for one predicate may diminish the value of a more uncertain predicate.

- **Lowest Scoring Predicate (LSP)**: The score for a sentence is the lowest score of all the predicate-argument structures present in the sentence. This strategy prioritises sentences that contain a predicate that is most likely to have a high impact on learning, although this may allow selecting for sentences that require annotating additional predicates that have already been learned well by the model.

In the case of our version of DO-BALD, a sentence with two predicates will have ten total forward-passes, five for each predicate. In the following example, a sentence contains one predicate that's very common and may likely already occur in the dataset, come.01 (*motion*), and a second predicate that's less common, make_it.14 (*achieve or arrive at*).

[$_{\text{ARG0}}$ The governor] [$_{\text{ARGM-MOD}}$ could] [$_{\text{ARGM-NEG}}$ n't] [$_{\text{make\_it.14}}$ make it] , so the lieutenant governor came instead .

The governor could n't make it , so [$_{\text{ARG1}}$ the lieutenant governor] [$_{\text{come.01}}$ came] instead .

A plausible scenario is that the predictions of the arguments for the rarer predicate "make it" will be in higher disagreement compared to the predictions of the arguments for "came". In this case, the LSP method will be more likely to select the sentence than AP, since it will rank this sentence's likely informativeness based only on the disagreement rate of "make it", whereas AP will average between the two disagreement rates.

3.3.2.3 Model Output

We also tested the classic approach of selecting query sentences based on the probability distribution over labels from the model's output. For each predicate in a sentence, we summed the highest probability value for each token and then normalised by sentence length. This results in a single confidence score for the label sequence. As with the BALD method, we must determine a score for the sentence itself, based on potentially multiple predicates. We test both the AP and LSP methods, either averaging the confidence for all predicates or using only the lowest.

3.3.2.4 Baselines

We include three passive baseline measurements:

- **Random Sentences (RandSent)**: Choose random batches of sentences on each iteration of active learning.
- **Random Documents (RandDoc)**: Choose random batches of entire documents, until the chosen sentence batch size is reached.
- **Most Predicates (MostPred)** Choose batches of sentences, selecting for those with the highest number of predicates present. Identification of predicates was done automatically using AllenNLP.

Sentences with a high number of predicates are very information-dense and often contain interesting lexical items and complicated syntax, which is why we included *MostPred* as a baseline. The downside is that because of their complexity, they typically take longer to annotate. The following is an example of a sentence with 5 predicates, denoted with underlining, which would be prioritised by the MP selection method:

> In an Oct. 19 review of "The Misanthrope" at Chicago's Goodman Theatre ("Revitalized Classics Take the Stage in Windy City," Leisure & Arts), the role of Celimene, played by Kim Cattrall, was mistakenly attributed to Christina Haag.

3.3.3 Results

Our results are reported as a learning curve across the number of sentences (Figures 3.1, 3.3) and predicates (Figures 3.2, 3.4) present in the training pool after each iteration. Selected F-scores for the methods are reported according to number of sentences (Table 3.2) and approximate number of predicates (Table 3.3) in the training pool at various points.

# sentences	300	600	900	1200	1500
OntoNotes					
RandSent	55.48	64.32	71.00	72.02	74.95
RandDoc	61.26	64.27	70.20	72.31	73.59
MostPred	59.39	**74.60**	**76.13**	**77.55**	77.52
DO-BALD LSP	60.25	73.48	74.80	76.23	**78.13**
DO-BALD AP	**62.26**	63.92	66.28	69.83	67.29
Output LSP	61.91	70.29	71.08	73.27	74.87
Output AP	62.12	58.52	64.52	62.28	68.39
THYME					
RandSent	64.53	72.07	74.23	75.67	76.88
RandDoc	49.32	64.23	67.11	73.62	75.21
MostPred	**66.66**	74.61	**76.37**	**77.49**	78.66
DO-BALD LSP	58.01	**74.66**	75.81	76.91	**79.03**
Output LSP	64.80	72.87	76.24	77.03	78.69

Table 3.2: F-score for number of sentences for each query selection method: random sentences, random documents, most predicates, DO-BALD (Lowest Scoring Predicate and Average of Predicates), model output (Lowest Scoring Predicate and Average of Predicates). Sentence count is approximate for whole-document selection.

Approx. # predicates	1000	1500	2000	2500	3000
OntoNotes					
RandSent	55.48	66.89	64.32	70.79	72.18
RandDoc	61.26	64.27	67.72	70.20	69.73
MostPred	-	-	59.39	-	-
DO-BALD LSP	60.25	68.27	68.26	**71.08**	**73.47**
DO-BALD AP	**62.43**	66.61	69.67	70.12	70.53
Output LSP	61.91	**68.83**	**70.29**	71.03	72.28
Output AP	56.68	56.00	62.28	68.39	71.09
THYME					
RandSent	66.47	72.06	72.25	**76.28**	75.67
RandDoc	64.23	67.11	73.32	75.35	**76.23**
MostPred	-	-	70.69	72.57	74.60
DO-BALD LSP	58.01	71.63	**74.66**	75.82	75.81
Output LSP	**67.30**	**72.87**	71.57	76.24	76.03

Table 3.3: F-score for approximate number of predicates for each query selection method: random sentences, random documents, most predicates, DO-BALD (Lowest Scoring Predicate and Average of Predicates), model output (Lowest Scoring Predicate and Average of Predicates). MostPred takes too large of selections to always be within range of these numbers.

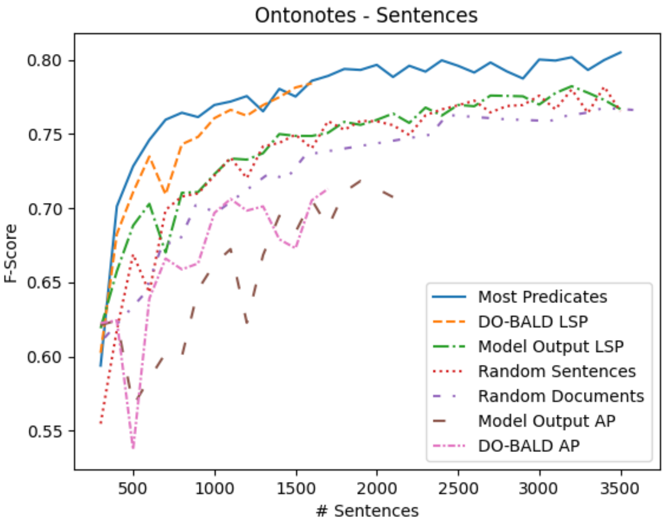


Figure 3.1: Learning curve of F-score by number of sentences in *OntoNotes* training data.

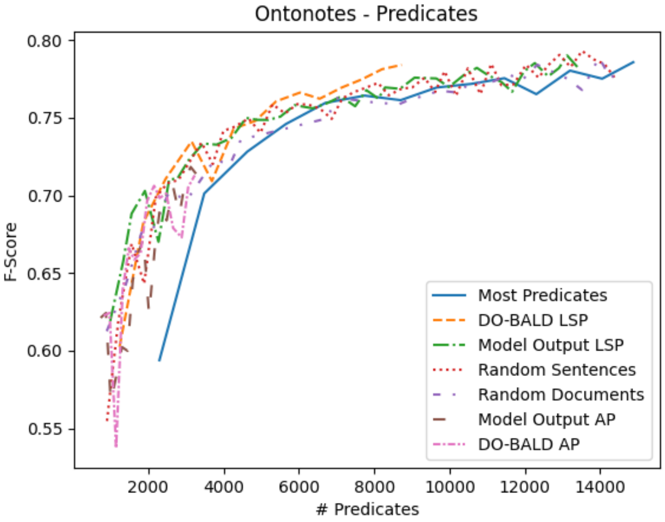


Figure 3.2: Learning curve of F-score by number of predicates in *OntoNotes* training data.

3.3.3.1 OntoNotes

We can estimate the annotation savings gained by the tested methods by examining the statistics required for each curve to reach a particular F-score. For this purpose, we will choose 78% as a benchmark of reasonable performance and around the point of performance plateau. This particular value was largely chosen for convenience, as most of our experiments were trained long enough to reach this score.

The passive selection of random sentences attains this score after 3,000 sentences. The DO-BALD LSP method and MostPred methods achieve this score after 1,400 and 1,200 respectively, providing a **53%-60% reduction in data**. Using the model's output with LSP provided a more slight, but still significant, reduction of 10%. When selecting whole documents, this performance was not achieved until 4,126 sentences were in the training pool. Both of the AP methods, which averaged the predicates in the sentences, performed significantly worse than the baseline. One contributing factor for performance degradation may be that the presence of frequent, but easily learned, predicates (such as copulas) inflates the average confidence of the sentence.

On the other hand, the reduction in *predicate* annotation offered by active learning was more modest compared to sentence reduction, but still substantial. The passive strategies of selecting random sentences and documents required 9,333 and 11,598 predicates, respectively. DO-BALD LSP required 7,673 predicates (18% fewer). The MostPred strategy, which offered the best performance on reducing sentences, didn't achieve this until 11,460 predicates, almost comparable to random whole-document selection. Output LSP provided a negligible reduction, with 9,073 predicates (3% fewer).

In terms of assessing the impact of whole-document selection, which is necessary for other NLP tasks such as coreference, compared to sampling random individual sentences, the difference between sentences (4,126 vs 3,000, respectively) and predicates (11,598 vs. 9,333) required to reach our benchmark was significant. Using random sentences rather than whole documents reduces sentence annotation by 27% and predicate annotation by 20% to reach our benchmark.

3.3.3.2 THYME Colon

Due to the weak performance of the AP aggregation method on the *OntoNotes* dataset, we did not perform those experiments on the *THYME Colon* dataset.

As with our evaluation on the *OntoNotes* dataset, we can consider the annotation requirements

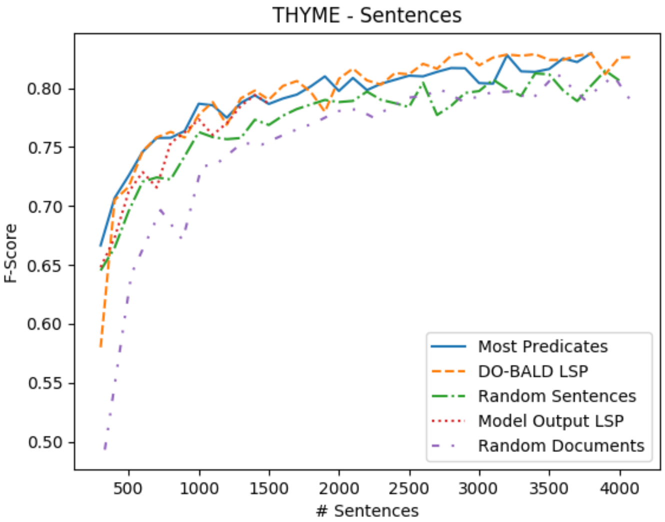


Figure 3.3: Learning curve of F-score by number of sentences in *THYME* training data.

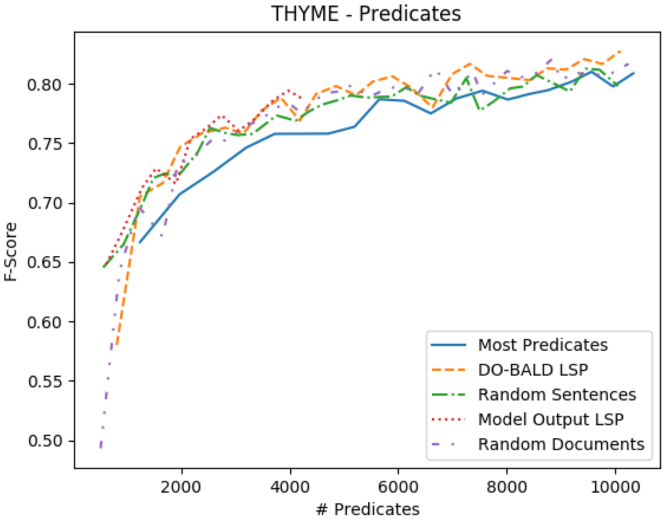


Figure 3.4: Learning curve of F-score by number of predicates in *THYME* training data.

to reach an F-score of 78.

The baseline sentence selection method obtains this benchmark after 1,600 sentences. Consistent with the results on the *OntoNotes* dataset, the DO-BALD LSP and MostPred methods are the most efficient ways of selecting sentences, with both requiring **60% fewer sentences to train a model with a test F-score of 78**. The Output LSP method requires 18% fewer sentences.

With respect to predicates, once again we see the baseline RandSent performance (4,355 predicates) significantly improved by DO-BALD LSP (20% less - 4,355 predicates) and Output LSP (16% less - 3,666 predicates), but MostPred is a detriment (30% *more* annotation - 5,651 predicates).

3.3.4 Conclusions

Between the two proposed methods of aggregating predicate-argument structure scores into a single value to represent a sentence, either averaging across them (AP) or only considering the weakest predicate (LSP), our results show the latter to be substantially better.

Both selecting sentences for the most predicates and selecting sentences with the predicate with the lowest DO-BALD agreement offer a significant 53%-60% decrease in the number of sentences required to train the model to a viable performance level with limited benefit to continuing. These findings are consistent for both the broad, general *OntoNotes* corpus and the niche colon cancer clinical note domain of the *THYME* corpus.

We assessed the performance of these selection strategies in terms of reducing both the number of sentences and number of predicates annotated. Typically, the SRL annotation process of a large annotation project benefits most from a reduction of predicates, due to presenting annotators with batches of a specific predicate to annotate, thereby reducing the cognitive load of switching between different predicate frames. However, in the case of projects attempting to develop new corpora with significant budget constraints that would most benefit from an active learning approach, the piecemeal nature of each annotation iteration makes this approach less viable and likely necessitates presenting annotators with the data sentence-by-sentence. In this case, reducing the number of sentences may be of more relevance.

While both DO-BALD LSP and the simpler strategy of selecting sentences with high predicate density provide significant reduction in sentence annotation, only DO-BALD LSP simultaneously reduced predicate annotation as well.

As previously described, Siddhant and Lipton [88] also used DO-BALD for SRL. Since they used a different model and slightly different dataset, we cannot directly compare our results, but the effect of AL appears to be more pronounced in our experiments. One key difference between our approaches that may have provided a benefit is that the authors treat agreement between predictions as all-or-nothing, whereas our calculation for agreement allows for partial agreement based on arguments. Additionally, we consider each predicate-argument label sequence independently in the case of LSP.

3.4 Predicate Selection

Since sentences in most domains typically contain multiple predicates, there are often redundancies in choosing predicates to annotate on the sentence level. Although a sentence may contain a particularly informative predicate, annotating high-frequency verbs such as “be” that co-occur in the sentence may not be beneficial. We instead use a method to select specific predicate-argument structures and compare the impact on performance as compared to selecting whole sentences instead.

This method is a natural extension of our previous experiments that allows us to even better leverage the focused annotation that active learning offers by using a more granular approach. While we find consistent early benefit in the more domain-specific corpora, this finer-grained approach proves to be slower for the more diverse *OntoNotes*.

3.4.1 Methods

As with the previous experiments, we use the *OntoNotes* and *THYME Colon* corpora, but also use the *Earthquakes* and *Ecology* corpora from *ClearEarth* (described in detail in Section 3.2.2).

We partitioned the training subset of each corpus into 200 random sentences for seeding the learner, with the remainder used as the initial “unlabeled” pool for selection. The initial 200 seed sentences were the same across the three selection methods tested for each respective corpus.

After initially training on the seed set, we then select a batch of either 100 predicates or a number of sentences that are comprised of approximately 100 predicates to add to the training pool using the BALD PREDICATES or BALD SENTENCES strategy described below in Section 3.4.1.1 or by choosing random predicates to simulate a passive learning approach. We evaluate the model on the test subset of the respective corpora and then the model is retrained with the extended training pool. We continue these iterations of selection and re-training until either all the data has been selected and moved into the training pool, or the experiment performances have sufficiently plateaued.

Our training procedure for this model used 25 epochs or stopped early with a patience of 5 based on the validation data for the relevant corpus.

3.4.1.1 Selection Methods

We use the most successful selection method of our previous experiments, the DO-BALD LSP method described previously in section 3.3.2. For clarity, in this section, we will refer to this method as BALD SENTENCES.

The new BALD PREDICATES method is a more granular extension of this previous work. We use the same idea of scoring individual argument spans based on agreement and averaging them into a single score for a given predicate instance, but we do not do the next step of combining the scores of all predicates within a given sentence. We instead use the score to choose specific predicate instances to add to the training pool.

We also compare these two active learning methods against a passive baseline of selecting random predicate instances.

3.4.2 Results

Natural variability in training the model produces some amount of noise, most prominently during the early iterations. In order to improve readability of these learning curves, we apply a Savitzky–Golay filter, which smooths the curves by fitting successive sub-sets of adjacent data points with a polynomial using linear least squares. We use a window of 15 data points and a cubic polynomial.

These learning curves are presented in Figures 3.5, 3.6, 3.7, and 3.8. We see consistent benefits of the BALD PREDICATES method at different points depending on the corpus.

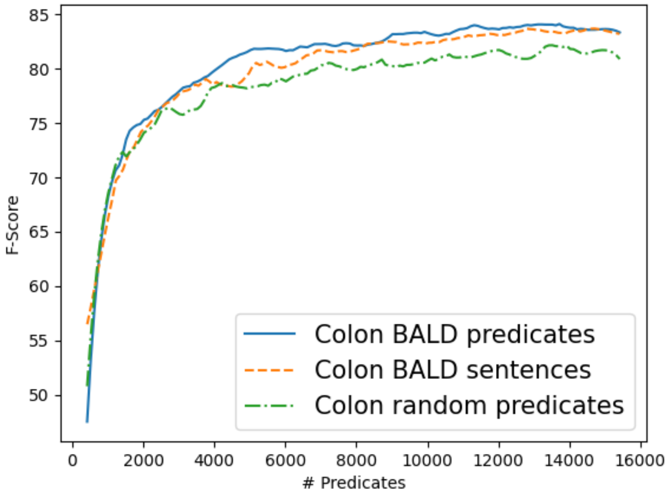


Figure 3.5: Performance of each selection method by number of predicates in the training pool on *THYME Colon* dataset.

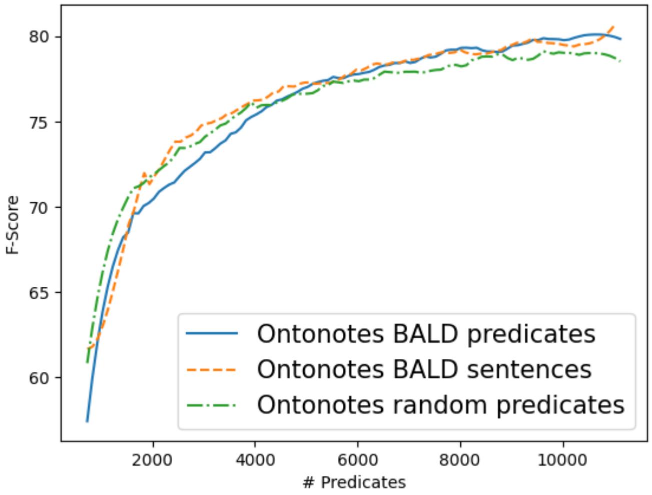


Figure 3.6: Performance of each selection method by number of predicates in the training pool on *OntoNotes* dataset.

For *Colon*, *Ecology*, and *Earthquakes* we begin to see consistent improvement for the BALD PREDICATES method over the other methods by approximately 1,500-2,000 predicates. On the other hand, for *OntoNotes*, it only catches up to random selection around 4,500 predicates and begins to improve over it around 7,000 predicates. For this corpus, BALD SENTENCES performs better.

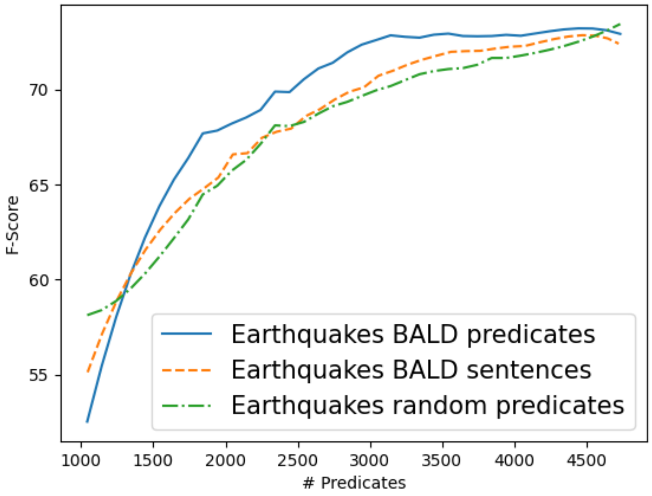


Figure 3.7: Performance of each selection method by number of predicates in the training pool on *ClearEarth Earthquakes* dataset.

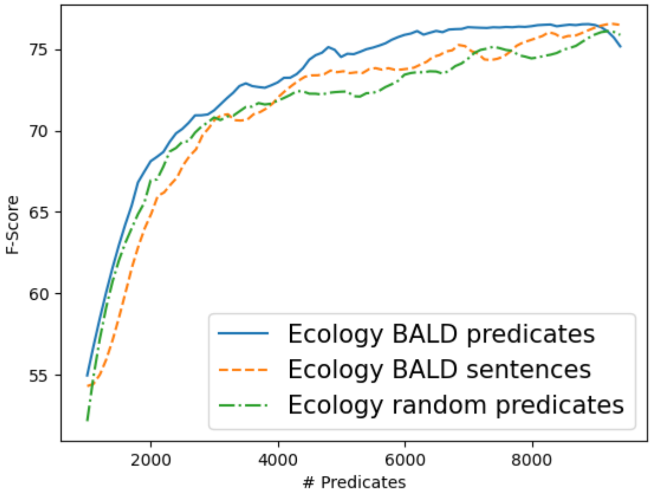


Figure 3.8: Performance of each selection method by number of predicates in the training pool on *ClearEarth Ecology* dataset.

3.4.3 Analysis of Selections

In order to better understand the differences between the selection processes used and their variance across datasets, we examine the selections within each batch.

3.4.3.1 Diversity

By selecting multiple predicates or sentences in each iteration, we expect that there may be redundancies. For example, if the model has never seen a given predicate, it will likely have low confidence in its predictions for it. We present a study of the diversity of the selections over time.

We first observe the amount of redundancy within BALD PREDICATES. This method is often choosing multiple instances of the same predicate lemma, as observed in Figure 3.9. In the two *ClearEarth* corpora we have analysed in this regard, which both ran to completion on the training data, approximately 25 of the 100 predicates chosen in a batch are duplicates in the early phase of active learning and with redundancy getting worse as the process gets closer to completion. The results for *Colon* contain approximately similar amounts of redundancy for the duration we trained it.

While there may sometimes be value in selecting the same lemma in order to obtain multiple senses of the same predicate, minimising this could prove beneficial. Future work could be done to study the effect of limiting the selection batch to unique lemmas.

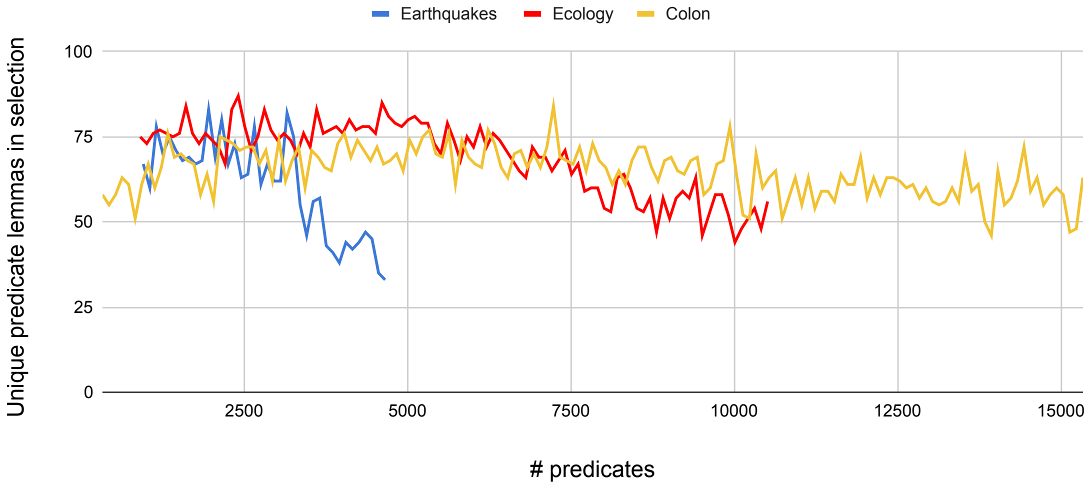


Figure 3.9: Number of unique predicate lemmas selected in each batch by the BALD PREDICATES method over iterations.

Additionally, the BALD PREDICATES method is capable of selecting multiple instances from the same sentence. While this may be beneficial, it's also possible that learning from just one predicate in the sentence will provide information that can improve agreement on other instances in the sentence.

We have found that for *Colon*, a randomly selected batch of 100 predicates contains 3 duplicate sentences on average, while the selections by BALD PREDICATES contain only 1 duplicate on average. For the *Ecology* corpus, both methods pick 3 duplicate sentences on average. This appears indicative that this is not a significant factor that necessitates correction.

Furthermore, we are interested in the sentence-level semantic redundancies within batches. Using the pre-trained all-mpnet-base-v2 model [92], we can calculate the average pairwise cosine similarity between the unique sentences within batches. In Figure 3.10, we find that both active learning methods contain more sentence-level similarity on average (0.26) than what is chosen through random selection (0.19) from the *THYME Colon* corpus.

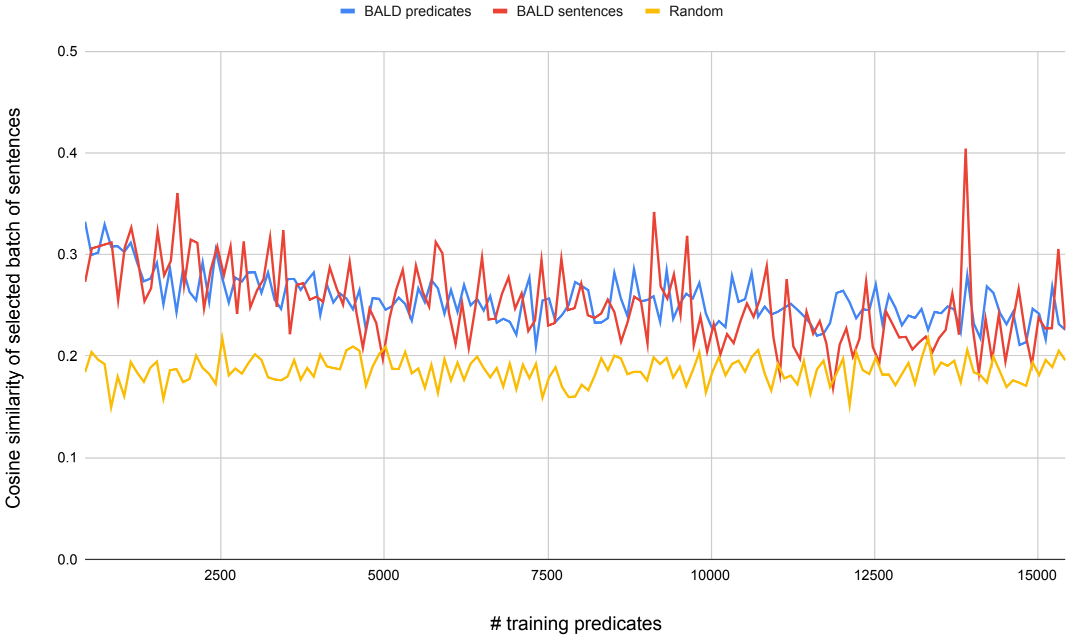


Figure 3.10: Average pairwise cosine similarity of selected sentences in each batch over iterations on *THYME Colon*.

We can see clear signs of the active learner choosing sentences that would be wasteful to have annotated. In one such batch, BALD SENTENCES selected 29 out of the 52 sentences where the sentences were all of the same basic form, but with varying cancer staging designations:

```
With available material : AJCC ypT1N0MX

With available surgical material [ AJCC pT3N2Mx ] .
```

On the other hand, the difference in selection diversity is less pronounced on the other datasets. In Figure 3.11, we show the similarity in the selections on *ClearEarth Ecology*, where all methods average 0.20 across the iterations.

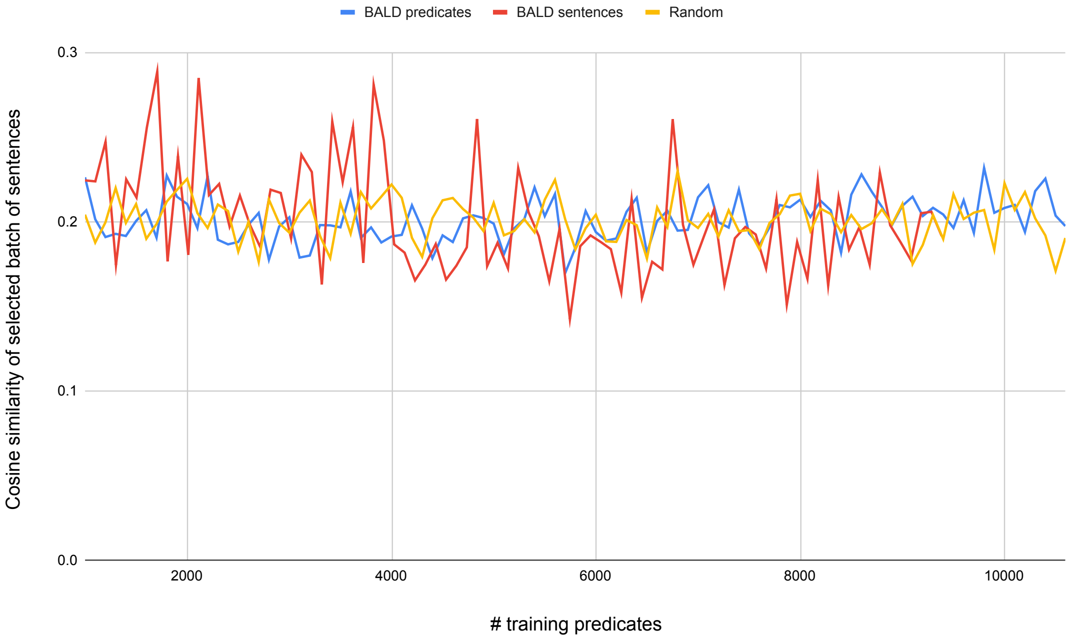


Figure 3.11: Average pairwise cosine similarity of selected sentences in each batch over iterations on *ClearEarth Ecology*.

Since the sentences chosen by the two active learning methods seem to have diversity that is reflective of the distribution of the training data, this is less concerning compared to the results on *THYME Colon*, but further reducing sentence similarity to below what we see in random selection could potentially be advantageous.

3.4.3.2 Vocabulary Coverage

We hypothesised that a contributor to BALD predicates's performance may be a rapid coverage of vocabulary, as predicates that involve unseen vocabulary could result in more disagreement. In Figure 3.12, we show the percentage of the unique vocabulary of the training set that is within the training pool as selections are made.

Across the datasets, we see varying results in how much BALD predicates expedites vocabulary coverage. We find that BALD predicates is not tending to choose unseen vocabulary compared to selecting predicates randomly for *Ecology*. On the other hand, active learning greatly accelerates this for *Ontonotes*, even after performance has largely plateaued. For *THYME Colon*, active learning provides an initial boost to vocabulary, but around the time that the performance plateaus, this decelerates below random.

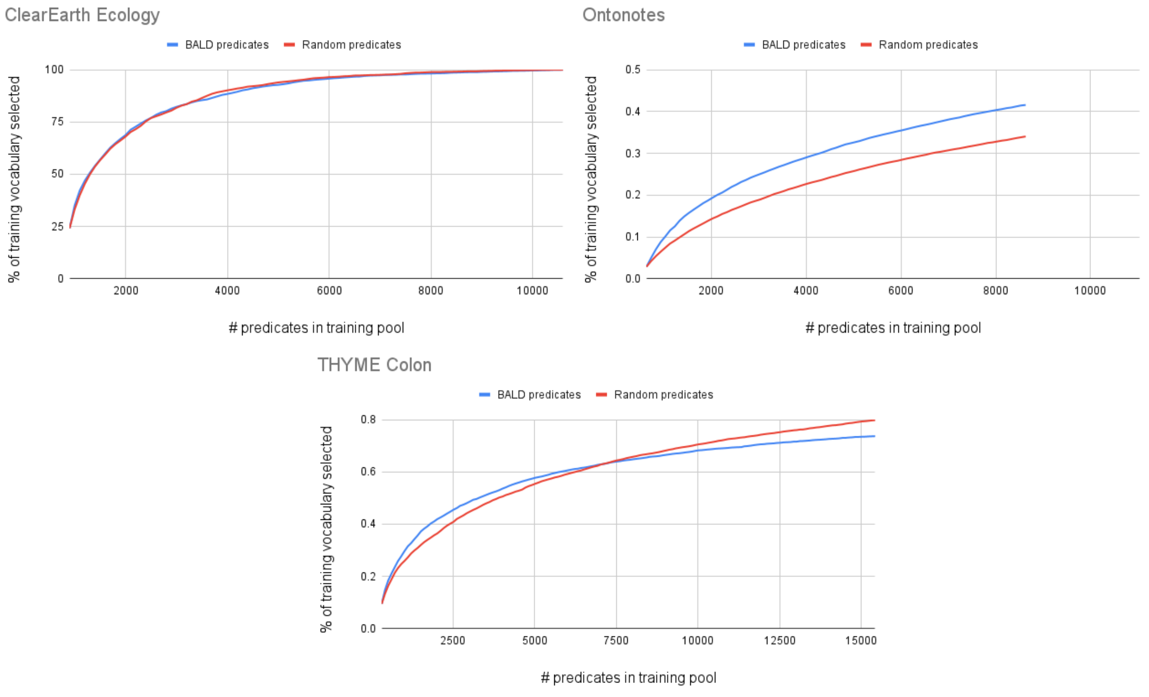


Figure 3.12: Percent coverage of training vocabulary in by number of predicates in training pool.

3.4.3.3 Disagreement

For BALD predicates, we calculate an average disagreement score for each selected batch. While early batches primarily contain predicates for which all predictions are in full disagreement, we see this disagreement trend downwards as performance plateaus. This is presented in Figure 3.13.

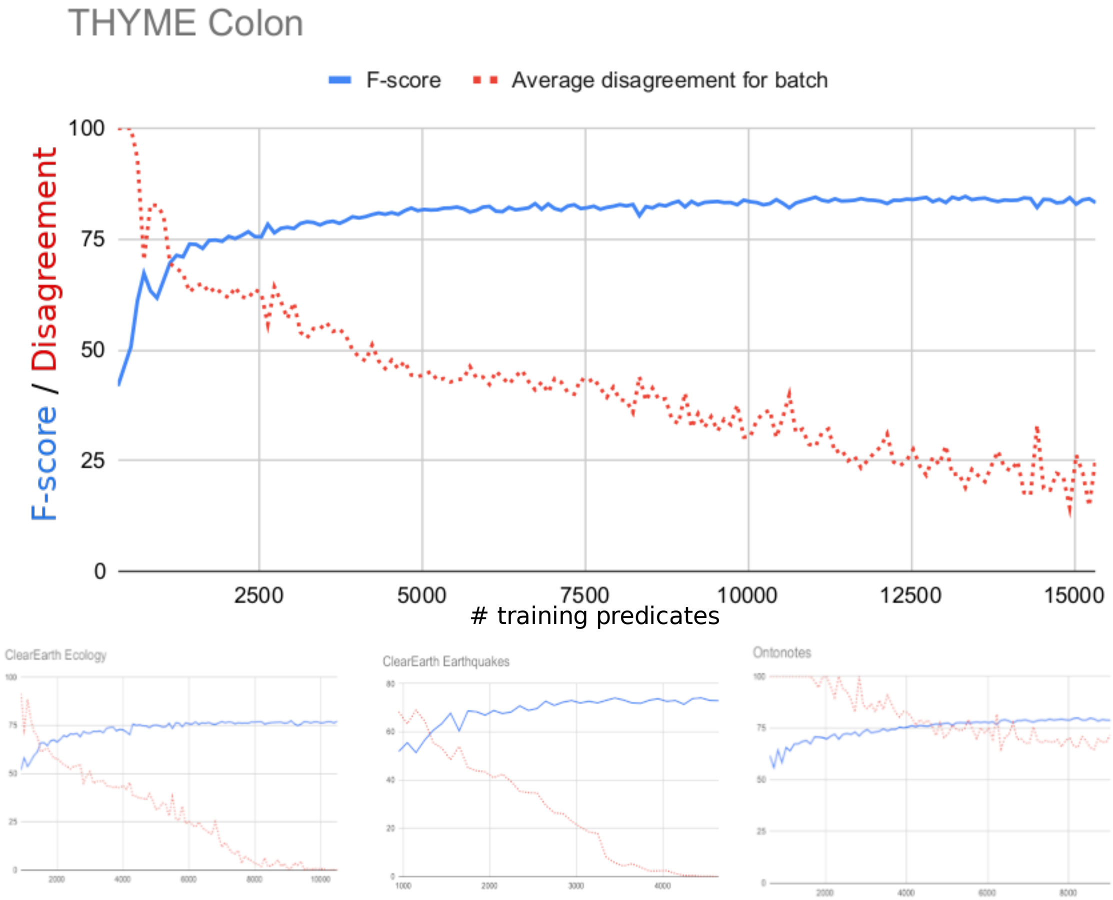


Figure 3.13: Average disagreement in selected batches decreases as iterations continue, while F-score increases and plateaus.

Although performance on *OntoNotes* has largely plateaued around an F-score of 79 by 7.5k training predicates, we know that training this model on the full dataset yields another 4 points. Since the disagreement scores of batches chosen by BALD predicates is still over 70%, this seems indicative of the additional further performance to be gained, albeit at a slow pace that gets little value for the effort. In contrast, *Colon* plateaued around 82, but the benefits of annotating the remaining 50k predicates only provides an additional increase of 1 point. With the disagreement score having fallen below 45%, this points toward an appropriate stopping point.

3.4.4 Corpus Analysis

Although the new predicate selection method offers immediate benefit over BALD sentences for the three sublanguage corpora, this is inconsistent with the result on *OntoNotes*, where selecting BALD sentences is more advantageous until about 7k predicates. In order to better understand the possible reasons for this, we compare the make-up and distribution of the corpora. These statistics are presented in Table 3.4.

We use PropBank roleset IDs as our measure of polysemy, since we have gold standard annotation for them in all 4 corpora. Note that PropBank sense distinctions are fairly coarse-grained and were generally only created when there were differences between senses with respect to the semantic roles. VerbNet [81], FrameNet [6] and WordNet [61] would all give much higher polysemy counts.

The largest and most diverse corpus in our experiments is *OntoNotes*, although we find that in terms of ratio of total tokens to predicates, unique rolesets, and unique tokens, *OntoNotes* is statistically more similar to the *THYME Colon Cancer* corpus than to either of the *ClearEarth* corpora. *OntoNotes* and *Colon* contain approximately one unique roleset per 376-403 tokens, whereas *Earthquakes* and *Ecology* contain one per 39 and 60 tokens, respectively.

Since *OntoNotes* covers a wider diversity of text types, it's unsurprising that it contains a much more diverse set of senses compared to the other corpora. While a lemma like "take" shows up with 25 different senses in *OntoNotes*, it only shows up in 8 senses in *Colon*.

For *OntoNotes*, only 30% of predicate occurrences are monosemous within the context of the corpus, whereas this figure is between 54%-61% for the other three corpora. A total of 6% of the unique predicate lemmas within *OntoNotes* are seen in 3 or more rolesets, while this is true of only 2% of the lemma types in each of the other corpora.

We believe this polysemy factor may contribute to the predicate selection method being disproportionately slower to improve the learning curve on *OntoNotes* compared to the more focused domain corpora. The model may be becoming overconfident after selecting many of the most common senses and failing to choose instances that would help it learn the remaining senses. BALD PREDICATES may be particularly disadvantaged, since BALD SENTENCES may still incidentally add these to the training pool.

	OntoNotes	Colon	Earthquakes	Ecology
Tokens	2.2 mil	522k	41k	83k
# tokens / # types	44.55	36.88	8.42	10.43
Predicates	301k	57k	7.5k	15k
Tokens per predicate	7.41	9.11	39.63	60.45
Avg sentence length	18.74	11.33	23.39	24.48
Unique rolesets	5535	1389	1046	1376
Tokens per roleset	403	376	39	60
Predicate lemmas with 1 roleset	3829 (83.33%)	1340 (90.24%)	985 (91.20%)	1416 (92.73%)
Predicate lemmas with 2 rolesets	494 (10.75%)	112 (7.54%)	73 (6.76%)	80 (5.24%)
Predicate lemmas with 3+ rolesets	272 (5.92%)	33 (2.22%)	22 (2.04%)	31 (2.03%)
Monosemous predicate occurences	29.95%	55.02%	53.53%	60.94%

Table 3.4: Statistics about the four corpora.

3.4.5 Conclusions

We've demonstrated that active learning can reduce annotation requirements for semantic role labeling across multiple domains by employing Bayesian Active Learning by Disagreement and using dropout to provide variability in predictions from the model. These predictions can be used to estimate the model's confidence in its predictions and select informative training instances to annotate.

Selecting predicate instances through the BALD PREDICATES method offers significant improvement in efficiency for *THYME Colon*, *ClearEarth Earthquakes* and *Ecology*, which have very focused domains. This method does not provide the same performance increase on the more general *OntoNotes* over the previous BALD SENTENCES, which selects whole sentences.

We have provided a statistical comparison of these corpora and offered a possible reason for the divergence in performance: a notable difference in polysemy within *OntoNotes* compared to the rest of the corpora.

Additionally, we examined the diversity of the selected predicates and sentences for BALD PREDICATES. Although these results vary across the different datasets, it indicates a potential avenue of future improvement. Reducing sentence-level semantic similarity seems of particular relevance to the *THYME Colon* corpus. We have also identified redundancies in the predicates chosen in each batch by BALD PREDICATES, which could be reduced in future work.

We also presented the change in model prediction disagreements over iterations as compared to model performance, which could be beneficial to determine when the costs of further annotation outweigh the additional gains that the model can provide.

3.5 Batch Sizes

Each iteration of active learning includes selecting an arbitrary number of instances to query. The number may be static, or dynamic with larger batches being selected in the early training process and smaller batches later on.

To maximally benefit from the model's feedback, in an ideal setup, each iteration would query for only one new instance, thereby minimizing the likelihood of selecting a batch of sentences with redundant information [80]. Unfortunately, this leads to the process of active learning being significantly slower due to needing to re-train the model more often. Additionally, annotating a sentence at a time with long breaks in between may cost additional time on the part of the annotator due to mental context-switching and needing to load up appropriate software and resources. It would be more efficient for them to be able to annotate numerous examples in a row.

Our previous experiments testing the BALD PREDICATES method show positive results when selecting 100 predicates in a batch. This small batch size requires about 60 iterations before the learning curve plateaus for the *Colon* corpus. We examine the effect of larger batches on the learning curves for the *THYME Colon* and the two *ClearEarth* corpora.

3.5.1 Results

We used the BALD PREDICATES selection strategy with varying sizes of 100, 500, and 1000 query instances. These results are presented for three datasets in Figure 3.14, using datapoints on intervals of 1000 predicates.

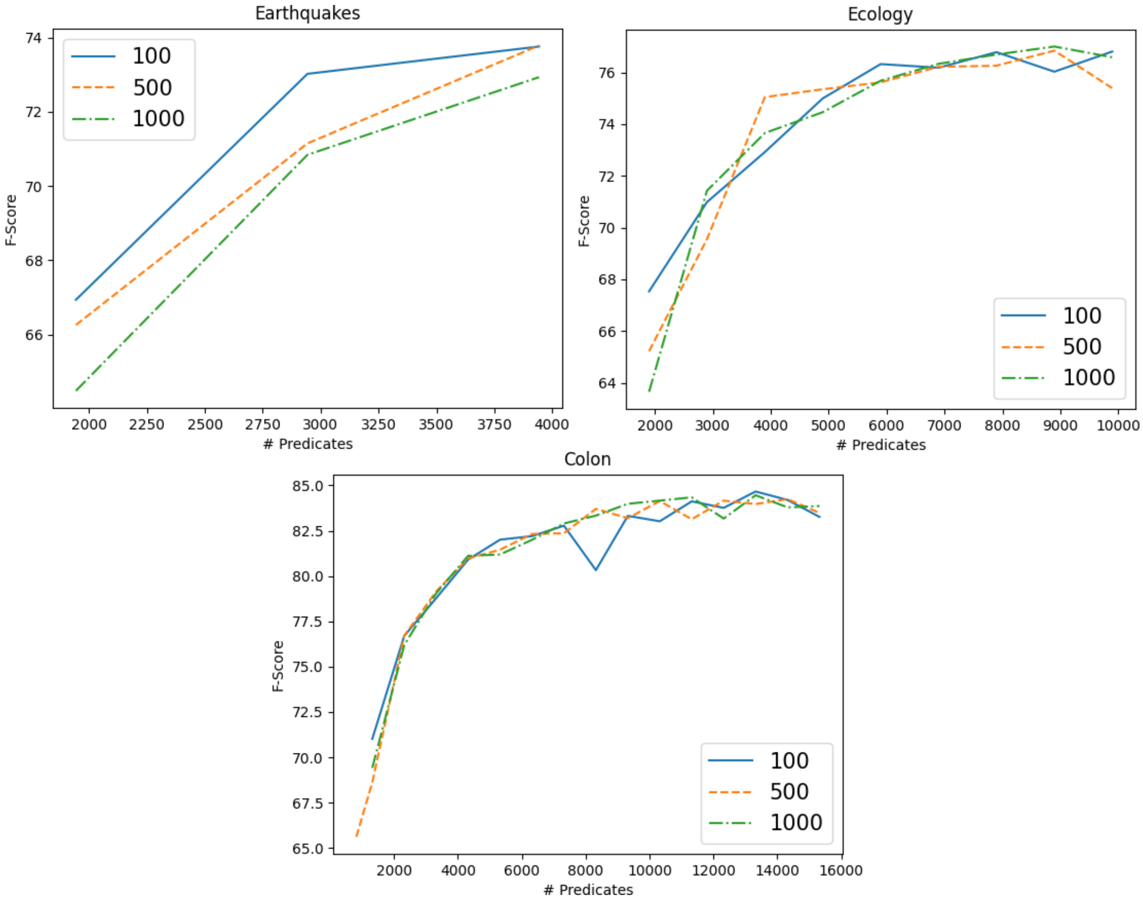


Figure 3.14: Performance of using BALD PREDICATES, selecting varying numbers of predicates per iteration.

Interestingly, changing the batch size has differing impacts on the datasets we examined this for. The *THYME Colon* corpus suffers very little from scaling all the way to 1000 predicates per selection batch. The results on *Earthquakes* show the clearest need for small batch sizes, while

Ecology exhibits shifting performance over the course of iterations.

3.5.2 Conclusions

Since the choice of how many selections to take on each iteration cannot be tuned for in real-world use of active learning, we have attempted to shed light on the levels of impact to expect on several different corpora, which vary in how sensitive they are to larger batches. We find that further investigation is needed to determine the most significant factors causing these differences so that future applications of active learning to SRL can predict the most ideal selection batch size that balances performance against training time for their target domain.

3.6 Seed Selection

Active learning requires a small number of labeled instances in order to initialise the model. Conventionally, these are chosen randomly, but this method can lead to issues. The majority of the popular active learning algorithms concentrate on selecting hard-to-classify instances that help quickly hone in on decision boundaries. This bias in sampling is what enables such rapid learning compared to passive learning, but the classifier may become overconfident about the membership of instances which belong to clusters from which it does not have labeled examples and misclassify them. If they lie far from a decision boundary, the algorithm may never select them for annotation. This phenomenon is called the *missed cluster effect*, and may result in entire classes being missed in imbalanced data sets where the seed set is chosen randomly. This special case of the missed cluster effect is called the *missed class effect* [98].

As mentioned previously, Dligach and Palmer [20] showed that a language model can provide good seed data for active learning for WSD by assigning low probabilities to rarer verb senses, which then allows for unsupervised selection of them to provide a better seed set. Peterson et al. [75] extended this to SRL, investigating the informativeness of low-probability instances for training an SRL model. The authors found that choosing those atypical sentences as the training data for an SRL model led to requiring fewer sentences for equivalent performance compared to choosing random sentences or the high-probability sentences.

3.6.1 Methods

3.6.1.1 Highest Perplexity

For the *THYME Colon* corpus, we trained GPT2 [78] on the training subset for 20 epochs. The model had an accuracy of 52.35% on the development subset. We used this model to rank the training sentences according to perplexity.

Since these clinical notes contain a significant number of formulaic sentences, simply choosing the sentences with the highest perplexity would lead to redundancies in the seed set, such as the following:

> Well - healed incision .
> Well - healed midline incision .
>
> # 10 Depression
> # 6 Depression

In order to prevent this, we applied an additional filter to ensure diversity in the seed set. We begin by choosing sentences in order of highest perplexity, but reject any sentences that have a cosine similarity higher than 0.85 with any of the sentences that have already been chosen.

We determined the sentences' embeddings based on the *Colon* GPT2 model combined with a pooling layer.

For *THYME Colon*, we selected 200 sentences (276 predicates) using this method as our initial seed data and performed active learning using the previously described BALD PREDICATES selection method. We also perform the same experiment using 200 sentences (648 predicates) of the *ClearEarth Ecology* corpus.

3.6.1.2 Most Dissimilar

We begin our seed set selection with one random sentence and proceed to iteratively select additional sentences. Using MPNet-base-v2 [1] to generate sentence embeddings, we calculate the pairwise cosine similarity between all candidate sentences that contain at least one predicate and the sentences chosen for the seed set so far. With the goal of finding another sentence to add to the seed set that is the most dissimilar to any already in it, we average the cosine similarity between each candidate sentence and each seed sentence. The most dissimilar sentence is added to the seed set and the process is repeated.

3.6.2 Results

In the case of choosing a seed with sentences with the highest perplexity, we present the learning curves for active learning on *THYME Colon* in Figure 3.15 and *ClearEarth Ecology* in Figure 3.16.

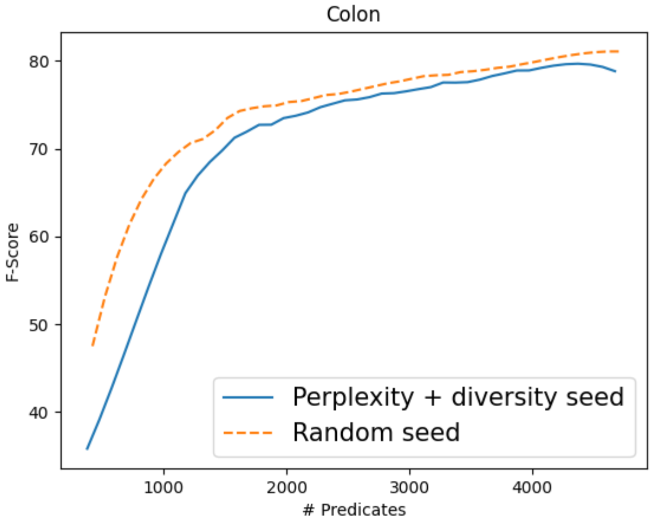


Figure 3.15: Performance of using perplexity to choose the 200 seed sentences vs. random on *THYME Colon*.

We find that using a seed set with the most unlikely sentences according to perplexity is

[1] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

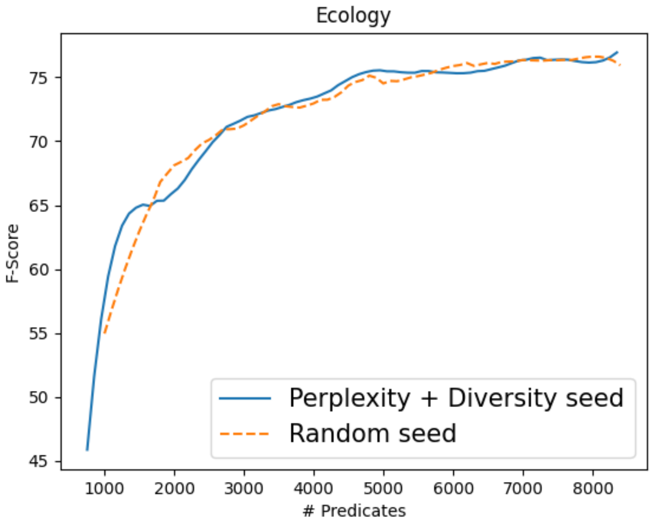


Figure 3.16: Performance of using perplexity to choose the 200 seed sentences vs. random on *ClearEarth Ecology*.

significantly detrimental to the performance of active learning on *THYME Colon*. For *ClearEarth Ecology*, we see an early gain, followed by a decrease in performance before the learning curve matches the version that started with a random seed.

Using the most semantically diverse sentences as the initial seed is far less harmful than using the perplexity method in the case of *Colon*, but does not appear to provide a benefit in either corpora, as seen in Figures 3.17 and 3.18.

3.6.3 Conclusions

We find that both selecting the seed based on high perplexity and using sentence cosine similarity to choose semantically distant sentences did not provide a performance benefit on the *Colon* and *Ecology* datasets. Although these methods were useful in previous NLP AL applications, there are many variables that differ between our experiments and prior work. Tomanek et al. [98] and Dligach and Palmer [20] were performing different NLP tasks, but Peterson and Palmer [75] found sentences with a low likelihood according to a language model to be more beneficial for SRL than an equivalent random sample. An important consideration though is that these previous

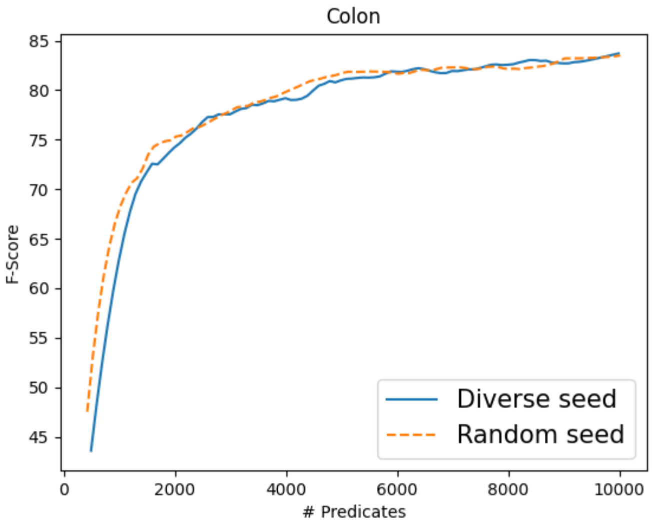


Figure 3.17: Performance of using 200 sentences with the least cosine similarity compared to each other as a seed vs. random on *THYME Colon*.

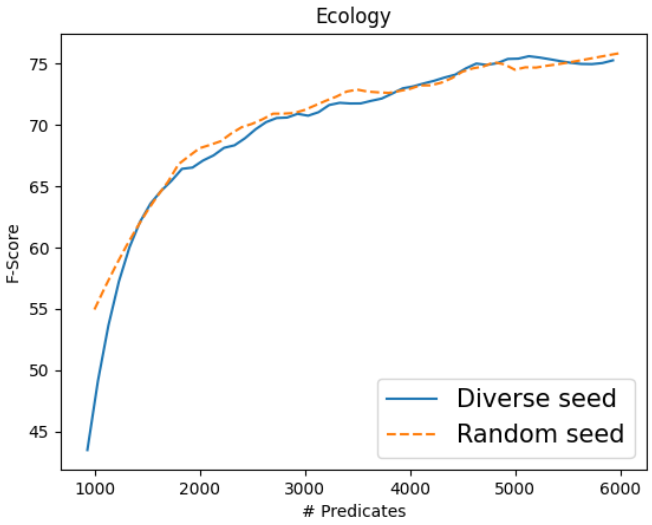


Figure 3.18: Performance of using 200 sentences with the least cosine similarity compared to each other as a seeds vs. random on *ClearEarth Ecology*.

experiments were not using neural nets and the much more robust modern embeddings from a large NN language model that we are. These advances in architecture and features may lead to the previously beneficial tuning of the seed set to become obsolete. Additionally, we tested these seed selection methods on narrow domains as opposed to more general texts, which may be less susceptible to the missed class effect due to their comparative lack of diversity.

3.7 Summary

In this chapter, we have discussed our experiments using active learning to accelerate the learning of an SRL model on new corpora. We estimate model uncertainty through Bayesian Active Learning by Disagreement, calculating the rate of disagreement of multiple model predictions, using dropout to provide variability.

We found that BALD does provide a more useful way of selecting new annotation instances for the SRL NN model compared to the model's output layer. By selecting sentences to annotate where the sentence contains a particularly interesting predicate that the model has high disagreement for, as opposed to averaging the agreement across all predicates in the sentence, we see even further benefit.

We also found that for datasets with sublanguage characteristics, it seems more beneficial to select individual predicate-argument structures to annotate and train on than to select all predicates in a sentence.

We found through examining the selections by BALD PREDICATES and random sampling that the comparative rates of vocabulary coverage can vary drastically between datasets. Although the amount of diversity within the batches of selections can vary between datasets, we have identified *THYME Colon* as a corpus that may benefit from introducing more sentence-level diversity. We see quite a bit of redundancy in the lemmas being chosen by BALD PREDICATES in all of the sublanguage datasets, which may also be useful to reduce.

We have presented a comparison of using active learning to select differing sizes of batches. While many of our experiments were performed using 100 sentences or predicates for each iteration, we find that increasing this to 1000 predicates causes negligible degradation in terms of performance on the *THYME Colon* corpus, while significantly reducing the time spent training the model and disruption to workflow. However, we find this result to vary based on the dataset. Further research is needed to identify the attributes of a particular domain that influence how detrimental larger batch sizes are.

We have found that creating seed sets using both sentences with a high perplexity according to a language model and sentences that offer a diverse coverage according to sentence embeddings fail to replicate the improvements seen in previous work for WSD and SRL. Since these prior findings utilised older models based on traditional ML, such as SVMs, we speculate that the improvements in SRL models and the word embeddings used as features have made these types of modifications less beneficial. Our earlier findings on the variations between targeted datasets may also account for the differences in our results, with the prior work on seed selection being performed on more diverse, general text.

Chapter 4

Bootstrapping AMR Parsing through SRL Active Learning

Since many SRL corpora do not have Abstract Meaning Representation annotations, in this chapter, we are investigating whether we can leverage existing SRL annotations using simulated active learning to provide more focused AMR annotation to efficiently develop AMR models for these domains. Due to PropBank frames being a vital part of AMR's logical representation, this overlap may be indicative that what is informative for an SRL model to learn is also informative for an AMR model.

4.1 Using BALD SRL for AMRs

As described previously in Chapter 3, model output probabilities are a less reliable measure of model confidence compared to using dropout as a Bayesian approximation for model certainty through successive forward passes. Unfortunately, using BALD as we have for SRL is not as straightforward for AMRs. Since AMRs are graph structures, it would require the development of an alignment algorithm to score disagreements. It may be possible to use an algorithm such as those used for SMATCH [14] to efficiently identify the correspondences between graphs, but we leave such experiments as future work.

4.1.1 Data

The *THYME Colon* corpus consists of clinical notes relating to colon cancer, as previously described in detail in Section 3.2.2.

We additionally test this method on a subset of *LORELEI*. The *LORELEI* corpus was previously described in Section 2.1.2. For both *LORELEI* and *THYME Colon*, we extract only the subset of training sentences that have both gold SRL and AMR annotations. In the case of *LORELEI*, this subset only consists of newswire text relating to disasters, and not the phrasebook, elicitation, or web forum text.

4.1.2 Methods

By performing AL using the BALD PREDICATES selection method (described previously in Section 3.4.1.1) on the subsets of these two datasets that contain AMRs, we can obtain a list of sentences in order of their priority to the SRL model. We initialised the active learning process with a starting seed of 200 random sentences from the respective datasets and select batches of 100 predicates. In the previous chapter, we found this method to increase the learning rate for the *THYME Colon* dataset. We verified that this same strategy also improves the SRL learning rate for *LORELEI*, which is shown in Figure 4.1.

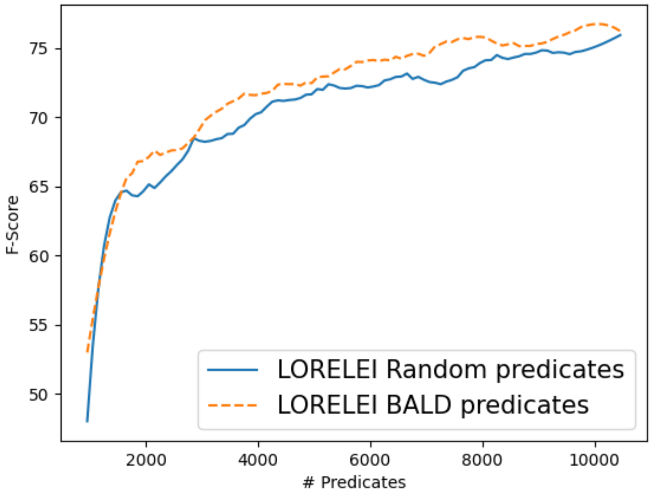


Figure 4.1: Learning curve of SRL by number of predicates in the LORELEI corpus for active and passive learning.

We created training sets from the first 1000, 2000, and 3000 gold AMR sentences that were deemed informative by the SRL active learner. We then trained the state-of-the-art SPRING AMR parser [8] on these increments, as well as trained a version of the model on an equal amount of random AMRs for comparison.

In the case of *THYME Colon*, we use a SPRING model that was pre-trained on the LDC AMR 3.0 release [44]. Since this release contains the *LORELEI* data, we train from scratch for those experiments.

4.1.3 Results

We report the SMATCH [14] scores on the test split for the training increments in Figures 4.2 and 4.3. We find that using the sentences chosen by SRL AL to train the AMR model performs significantly worse than using random sentences. On the *THYME Colon* corpus, performance is reduced by 1-5 points. The effect is more deleterious for *LORELEI*, reducing performance by 2-6 points.

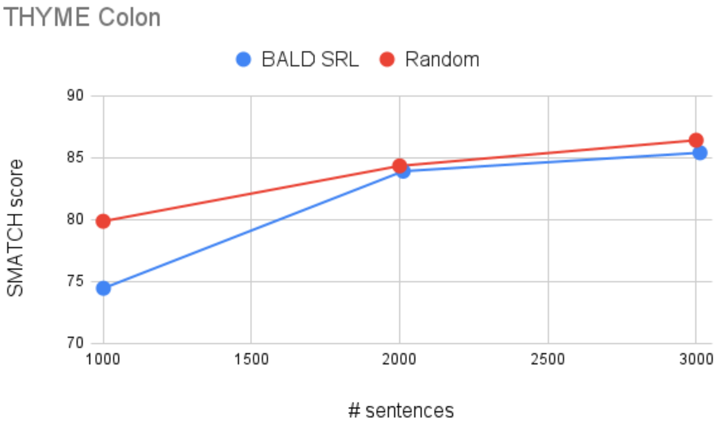


Figure 4.2: Comparison of training the AMR parser on random sentences vs. the sentences chosen by SRL AL for the THYME Colon corpus.

4.1.4 Conclusion

Despite AMRs containing SRL as a subset of their representation, we find that using the sentences that boost an SRL model's performance does not translate to improving the AMR parser's

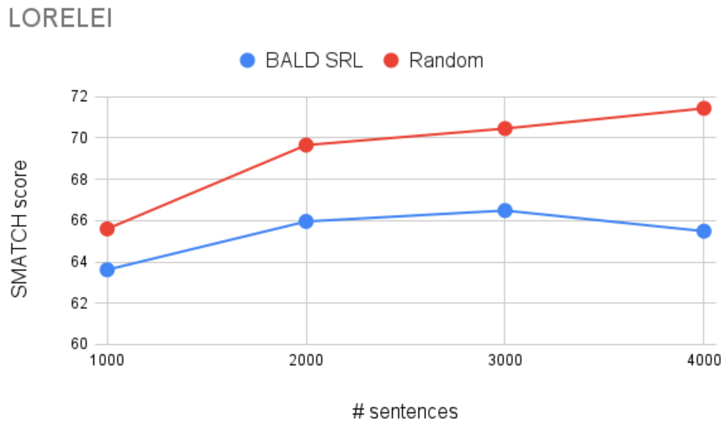


Figure 4.3: Comparison of training the AMR parser on random sentences vs. the sentences chosen by SRL AL for the LORELEI corpus.

performance. These sentences chosen by BALD for SRL are detrimental compared to using random sentences.

One reason for the difference may be due to the SRL active learning being biased towards picking sentences that have predicates in them. A sentence with no predicates tends to produce fewer disagreements over multiple predictions since identifying a sentence as not containing a predicate is quickly learned. The *THYME Colon* corpus is a unique domain where the author frequently drops predicates in favour of shorthand. Additionally, many of the formulaic sentences are lacking in predicates, such as listing medications being taken or vital signs. While these sentences are not highly valuable for training an SRL model, they are informative for training an AMR parser and should not be neglected. We can see this in the following example from the vital signs section:

```
Temperature = 98.78 [ degF ]
(h / have-quant-91
   :ARG1 (c / clinical-attribute
      :name (n / name :op1 "temperature"))
   :ARG2 (t / temperature-quantity
      :quant 98.78
```

```
    :scale (f / fahrenheit)))
```

Future work could explore using a random subset of these types of sentences to ensure coverage.

Furthermore, AMRs also may contain implicit events. Consider the following sentence:

```
Fluorouracil [ ADRUCIL ] solution 4,810 mg intravenous [...]
(t / therapy-01 :implicit +
    :ARG2 (m / medications-drugs
        :name (n / name :op1 "fluorouracil")
        :ARG1-of (d2 / dose-entity-91
            :ARG2 (s / solution)
            :ARG3 (m2 / mass-quantity
                :quant 4810
                :unit (m3 / milligram))))
    :manner (i2 / intravenous-01
        :ARG1 m))
```

Not only does the AMR parser need to learn that Fluorouracil is a *medications-drugs* entity, but also that there is an implicit *therapy-01* event in this context.

However, since the *LORELEI* newswire text has very few sentences without predicates and no :implicit events, the drop in performance cannot be solely explained by this phenomenon. Further analysis of the differences between the types of sentences being chosen by BALD and the composition of the corpus may shed light on these results and raise potential strategies of mitigating the problem.

Another consideration is the extent of the differences between SRL and AMR representations. In order to successfully predict the following example from *LORELEI*, the AMR parser needs to be able to extrapolate that “one of those” refers to a single *person*, the presence of the *include-91* frame, as well as that Hong Kong is a *city* named entity. PropBank doesn’t need to do these additional steps when it can simply label “one of those listed” as an ARG1 of the *miss.01* predicate.

```
One of those listed missing is from Hong Kong [...]

(b / be-located-at-91
 :ARG1 (p / person :quant 1
    :ARG1-of (i2 / include-91
       :ARG2 (t / that
          :ARG1-of (l / list-01
             :ARG2 (m / miss-01
                :ARG1 t)))))
 :ARG2 (c / city :wiki "Hong_Kong"
    :name (n / name
      :op1 "Hong"
      :op2 "Kong")))
```

4.2 Summary

In this chapter, we investigated whether existing SRL annotations can be used to inform AMR annotation through the use of simulated Bayesian Active Learning by Disagreement.

Despite sharing predicate frames, we have not found using BALD SRL to choose sentences for AMR training data to be more efficient than choosing random sentences to annotate AMRs. The extent of the differences between the representations may simply be to great for this strategy to work. Future work could test using conventional active learning based on the AMR model's output probabilities.

Chapter 5

Annotation Projection

Besides using active learning, we are also interested in using annotation projection techniques to develop new SRL corpora. As we described in Chapter 2, this approach utilises parallel text to leverage semantic annotation in a high-resource language to create automatic annotations in a target language.

In Section 5.1, we continue the review of the previous literature on annotation projection, which was started in Chapter 2, focusing on the aspects that relate to error analysis and filtering approaches.

In Section 5.2, we will present our preliminary results projecting SRL annotations from English to Russian using the *Russian PropBank 2020* subset [63] of the *LORELEI* dataset, and examine the errors that result from our projection methods.

These findings have led to additional annotations and corrections resulting in a new version of the dataset, *Russian PropBank 2023*. In Section 5.3, we describe these recent updates to Russian PropBank and compare it to English PropBank.

In Section 5.4, we use the Universal PropBanks 2.0 system to project to the updated *Russian PropBank 2023*. We evaluate this state-of-the-art annotation projection system and identify additional filtering methods, both specific to Russian and applicable broadly, to incorporate into the system.

5.1 Background

As described in Chapter 2, annotation projection requires the identification of correspondences between parallel sentences through unsupervised word alignment models or multilingual embeddings. Because of errors and imprecise translations, these alone are not sufficiently accurate to create high quality corpora in target languages.

As with other unsupervised models, word alignment models suffer from considerable noise. Even the most recent word alignment methods result in significant error rates, as we show in our comparison in Section 5.2.2. In order to reduce the issue of these errors propagating into the semantic projection, researchers have proposed numerous ways of either deterministically filtering out poor alignments or using an SRL model to smooth the errors.

Akbik et al. [2] performed a detailed analysis of the types of errors they encountered in the process of projecting annotations into French for the creation of the Universal PropBanks. Many of the errors encountered were due to translation shifts between the languages. A common error with projecting the predicate occurred when the target verb is simply not a semantic equivalent to the original source English verb. This type of error they corrected by using a translation dictionary to verify the validity of a predicate alignment. Another common error was aligning an English verb with a non-verb in the target language, which is indicative of a translation shift. These cases can be filtered by checking that the POS of the target word is also a verb.

Additionally, they found errors where the projection target was not the syntactic head of the complement. Rather than removing these projections as mistakes, they corrected them by moving the label to the nearest node with a verb as its immediate ancestor. Although these techniques all improved precision, even as much as from 45% to 88% for predicates and 43% to 75% for arguments, there still remain errors where the English frames are simply incompatible with the target language. This may be either due to there being no semantic equivalent in the target language or target language-specific syntactic particularities. Since the authors later remove sentences with incomplete annotations, they did not prioritise methods to improve recall.

Fei et al. [29] used fast_align to provide word alignments for their annotation projection from English PropBank to automatically translated parallel sentences. Each potential projection was given a score, calculated by a joint probability between word alignment probability and the predicted part-of-speech probability distribution. They tested multiple values of a threshold hyperparameter for weeding out low likelihood projections, determining that F1 was most improved by ignoring projections under the score of 0.4, with the score diminishing if the parameter was set higher.

The authors of X-SRL [19], which we briefly described in Chapter 2, also filtered candidate projections. They kept only targets which had a verbal POS tag, which had the effect of removing light verbs. Since alignments were to *word pieces* in the target sentence using mBERT embeddings, those were used to 'vote' on the correct target word. If they still found multiple potential projections for a source word, they kept only the one with the highest similarity score. Compared to using only cosine similarity to perform projection, these filtering methods raised F-scores for both the method of projecting with alignment intersections or just source-to-target. The F-score on the latter increased by 4.5-9.2% to 76.9, 85.9, and 81.2 on German, Spanish, and French, respectively.

5.2 Projecting English PropBank to Russian PropBank 2020

We present our preliminary work on word alignment-based projection methods for English-to-Russian SRL as a baseline for future research. We compare the performance of several word alignment systems on a small manually aligned subset of the test set. The best performing system is used to provide word alignments for projection. We analyse the discrepancies between the projected annotations and the manual annotations, and then compare several filtering and correction techniques to reduce these errors.

In Section 5.2.5, we perform error analysis to identify 1) where gaps and inconsistencies exist in the Russian PropBank annotation that need correction, 2) systemic errors that the projection makes due to English bias, 3) idiosyncratic errors caused by the projection that may be improved by better filtering or word alignments.

5.2.1 Data

As previously described in Chapter 2, the Low Resource Languages for Emergent Incidents (LORELEI) project [1] released parallel corpora for numerous low resource languages, including Russian. The dataset includes sentence-level alignment between the English and Russian that can facilitate projection. A subset of the Russian data, consisting of *newswire* and *phrasebook* sentences, was annotated with PropBank-style semantic roles [63], referred to as Russian PropBank. The *phrasebook* subsection comprises relatively short sentences with highly parallel (though often idiomatic) translations, whereas the *newswire* subsection consists of complex, paraphrased sentences that are parallel only at the sentence level. A typical *phrasebook* sentence in English is “Go to bed”, while a typical *newswire* sentence may not only include multiple clauses or quotations, but is more likely to contain predicates of lower frequency, such as *subside.01* or *crisis.01*, which are unlikely to appear in the *phrasebook* section. This complexity and broader vocabulary make the *newswire* section a much more challenging dataset to project to.

This version of Russian PropBank consists of frames for 96 verbal lemmas. A portion of the LORELEI corpus was double-annotated and adjudicated with these frames, but coverage of the remaining predicates, including all predicate adjectives and nominalisations, were added later by a single annotator. Since official frames were not constructed for these additional predicates, the arguments were chosen to follow the general format of ARG0 being the prototypical agent, ARG1 being the prototypical patient, etc.

As part of preparing our data for projection, we automatically moved the span-based English annotations to the headwords. We used UDPipe [94] with a pre-trained model to provide a universal dependency parse and part of speech tags for the Russian text, which will be used for filtering the projections.

Additional datasets used in this work are UMC 0.1 [43] and UMC003 [46]. These are news articles in Czech, English, and Russian, comprised of 97,000 and 2,750 English sentences, respectively. These datasets used automatic alignment to provide parallel sentences. We use the English-Russian

[1] https://www.darpa.mil/program/low-resource-languages-for-emergent-incidents

sentences to provide additional training data for the word alignment models.

5.2.2 Word Alignment

As discussed in the previous Background section, projection methods depend heavily on unsupervised word alignments. In order to compare different word aligners and set-ups, we single-annotated the alignments of a small subset of the LORELEI data (109 sentences), consisting of both newswire and phrasebook sentences.

We compared several configurations of off-the-shelf word alignment systems. Two of these, fast_align [26] and *efmaral* [71], are extensions of the IBM models proposed by Brown et al. [12]. We trained these aligners on the parallel newswire, blog, forum, and phrasebook English-Russian data from LORELEI and English-Russian portion of UMC 0.1 and UMC003. We compared these with two recent alignment methods based on embedding similarity: AWESOME [22] and SimAlign [37]. In both cases, we used the standard pre-trained multilingual cased BERT embedding model. The metric used to evaluate the aligners was Alignment Error Rate (AER), where 0 would be perfect alignment accuracy. Our results are reported in Table 5.1.

Aligner	**Training Data**	**AER**
fast_align	LORELEI	28.57%
fast_align + lemm.	LORELEI	26.78%
efmaral	LORELEI	19.62%
efmaral + lemm.	LORELEI	17.33%
efmaral	LORELEI + UMC	16.33%
***efmaral* + lemm.**	**LORELEI + UMC**	**15.64%**
SimAlign mwmf	mBERT	23.51%
SimAlign mwmf + lemm.	mBERT	24.78%
SimAlign inter	mBERT	19.50%
SimAlign inter + lemm.	mBERT	20.08%
SimAlign itermax	mBERT	19.97%
SimAlign itermax + lemm.	mBERT	21.29%
AWESOME	mBERT	19.27%
AWESOME + lemm.	mBERT	20.60%

Table 5.1: Comparison of performance of word alignment setups on aligned Russian-English LORELEI data subset.

The performance of *efmaral* significantly exceeded the performance of fast_align, and slightly exceeded the two embedding-based word alignment systems. Since Russian is a language with very rich morphology, word aligners such as *efmaral* and fast_align may have difficulty learning alignments for words the occur infrequently in the training data. As discussed in Borisov et al. [11], automatic word alignment for English-Russian can be improved by lemmatising the Russian text. We found lemmatisation to improve AER by approximately 2% on both IBM-model-based aligners. Possibly because the mBERT-based aligners don't suffer from the same data sparsity issue, using lemmatisation decreased performance by approximately 1%.

The best performance of the approaches we tested was from *efmaral* trained on the LORELEI and UMC corpora with the Russian text lemmatised. We used this configuration to perform the projection methods described in the next section.

5.2.3 Projection Methods

By using the automatic alignments provided by *efmaral* trained on LORELEI and UMC, we were able to map the manual annotations from English to specific words on the parallel Russian sentences.

Since Russian PropBank does not include annotations for predicates without a verb alias, we do not project any of the 1,700 English roleset IDs that belong to predicates where none of the rolesets have a verb alias. This would include eventive nouns such as "cyclone" as well as stative adjectives such as "good".

We also did not project the particles in discontinuous verb-particle constructions (labels with a 'C-' prefix), as these tend to be highly specific to English and would not map appropriately. For example, we ignore the particle *out* in the following verb particle construction:

Did [you $_{\text{ARG0}}$] [let $_{\text{let.01}}$] the [dog $_{\text{ARG1}}$] [out $_{\text{C-ARG1}}$] ?

The automatic word alignments sometimes align one word in the source language with multiple words in the target language, or multiple words in the source language to the same word in the target language. In the case of an English word going to multiple Russian words, we chose to project to only the earliest occurring word. In the case of multiple English predicates being aligned with the same Russian word, we chose the first predicate to occur. In both cases, the choice was made only after any relevant filtering rules were applied.

Since the automatic alignments do contain inaccuracies, we compared several simple ways of filtering egregious errors. We first provide a baseline without any filtering:

Direct projection: The most basic form of projection is based on word alignments. Given two parallel sentences in the source language SL and target language TL and a list of word alignment pairs of the form $(SL_i, TL_{i'})$, we transfer the predicate label from SL_i and $TL_{i'}$ if there exists a word alignment between SL_i and $TL_{i'}$ and no previous predicate has yet been transferred to $TL_{i'}$. The semantic relationship $R(SL_i, SL_j)$ is transferred to $R(TL_{i'}, TL_{j'})$ if there exists a word alignment between SL_i and $TL_{i'}$ and between SL_j and $TL_{j'}$.

We test four methods of projection filtering, as well as using all of them in tandem:

Reattachment heuristic: Described by Akbik et al. [2], this filtering method aims to reduce errors caused by the argument in the target language not being the syntactic head. We implement a slightly modified form of this. If a candidate argument is not the direct child of the verb (or linked via *advcl*, *xcomp*, or *dep* dependency labels), we ascend the tree and place the argument on the immediate descendent. Arguments are not projected if the verb is not found in its ancestry. We do not apply this filter when the Russian predicate is not a verb or when projecting ARGMs.

POS filtering: Filter predicates and arguments if the proposed target is a PART, PUNCT, PRP, or ADP. We allow an exception for PART if the argument label is ARGM-NEG or ARGM-DIS.

Russian-specific heuristics: We make systematic adjustments according to common translation conventions and incompatibilities between English and Russian frames.

Although enumerating a thorough list of English-Russian semantic divergences would require substantial time and expertise, we created a short list of rules to correct for frequent mistakes caused by significant mismatches between English and Russian frames. We filter the following rolesets:

- Modal **have.02** (e.g. *"I have to go"*) is typically translated to a form of the adjective *должен*, which is not a predicate in the Russian dataset.

- Auxiliary rolesets **have.01** (*"Have you seen it?"*), **do.01** (*"Don't leave"*), and **be.03** (*"What are you doing?"*). The latter does have a parallel in Russian that was not annotated in this version of Russian PropBank.

- The roleset **need.01** is typically translated to an adjective (*нужно*) or adverb (*надо*) in Russian, which are not marked as predicates in the Russian dataset. Since verbal predicates do exist (*нуждаться*), we filter this predicate only in the case where the target POS is not a verb.

- Present tense *быть* (*to be*) is frequently implicit in Russian, and the word aligner often mistakenly aligns the English with the wrong word in Russian. Since this verb is so common in English, we include a rule that checks if the target argument for *be.01* (copula) or *be.02* (existential) is a form of *быть*, and filters it if it is not.

Ground-truth fallback: This method attempts to improve recall by using a bilingual dictionary to find the corresponding target word when the automatic word alignments fail to find a match in the Russian sentence. Since this doesn't give us which occurrence in the sentence matches with which word, we do not project the annotation if there are multiple potential matches. In our experiments, we use MUSE's English-Russian bilingual dictionary [50], which contains 53,186 word pairs.

All methods: We combine all of the above techniques together.

5.2.4 Results

Tables 5.2 and 5.3 show the respective results for the *newswire* and *phrasebook* sections of Russian PropBank. Predicates are evaluated as unlabeled, that is, a predicate assigned to the same token in the predicted data as in the test data evaluates as correct regardless of roleset. The first three columns show this result. The tables also show scores from evaluating arguments,

	Predicates			Arguments of correct predicates (unlabeled)			Arguments of correct predicates (labeled)			All arguments (unlabeled)			All arguments (labeled)		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Direct	75.4	77.7	76.5	76.3	78.0	77.1	62.4	63.8	63.1	75.9	**71.5**	73.6	59.5	**56.2**	57.9
Reattachment	75.4	77.7	76.5	80.6	74.5	77.4	65.6	60.6	63.0	78.3	67.6	72.6	61.3	52.9	56.8
POS filter	83.9	**78.1**	80.9	78.3	77.2	77.8	63.9	63.1	63.5	80.7	69.2	74.5	63.8	54.7	58.9
Russian heuristics	89.1	75.8	81.9	76.3	**79.0**	77.6	62.6	**64.8**	63.6	80.7	69.7	**74.8**	63.7	55.0	**59.9**
Ground-truth	76.3	77.1	76.7	76.1	78.6	77.3	62.5	64.5	63.5	76.2	71.1	73.6	60.0	56.0	57.9
All methods	**91.1**	75.8	**82.8**	**81.3**	76.0	**78.6**	**66.0**	61.7	**63.8**	**85.2**	65.0	73.7	**67.0**	51.1	58.0

Table 5.2: Results on the *phrasebook* portion of the Russian PropBank. Argument scores are calculated for only the predicates that the projection labeled correctly.

	Predicates			Arguments of correct predicates (unlabeled)			Arguments of correct predicates (labeled)			All arguments (unlabeled)			All arguments (labeled)		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Direct	72.8	**60.8**	66.2	47.6	53.5	50.4	36.0	40.5	38.1	59.3	**52.3**	55.6	42.3	**37.3**	39.6
Re-attachment	72.8	**60.8**	66.2	58.5	52.8	55.5	42.3	38.1	40.1	66.9	48.8	56.4	47.1	34.4	**39.8**
POS filter	79.0	**60.8**	**68.7**	48.9	53.3	51.0	37.2	40.5	38.8	62.7	50.0	55.6	44.1	35.1	39.1
Russian heuristics	79.4	59.0	67.7	48.0	55.0	51.3	35.8	41.1	38.3	59.4	50.8	54.8	41.9	35.8	38.6
Ground-truth	73.1	58.7	65.1	48.7	54.1	51.3	37.1	41.1	39.0	60.7	51.4	55.7	43.3	36.7	39.7
All methods	**84.5**	57.2	68.2	**59.0**	**56.7**	**57.8**	**43.1**	**41.4**	**42.2**	**70.9**	47.1	**56.6**	**49.4**	32.8	39.4

Table 5.3: Results on the *newswire* portion of the Russian PropBank. Argument scores are calculated for only the predicates that the projection labeled correctly.

filtering out arguments that belong to predicates that the method did not label correctly. It should be noted that this results in evaluating on different sets of arguments between the methods, so scores are not strictly comparable. Scores are presented with and without the argument label itself (ARG0, etc.) being taken into consideration. The final six columns contain scores on all dataset arguments, showing the overall coverage that these methods achieve. We also present confusion matrices showing the argument errors for the direct projection in Tables 5.4 and 5.5.

The results on the *phrasebook* section consistently surpass those on the *newswire*, which can likely be attributed to the difference in length of sentences, breadth of vocabulary, and how parallel the sentences are on a lexical level. Using all of the filtering methods improves predicate identification from 76.5 to 82.8 F-score on *phrasebook*, but the *newswire* section only increased from 66.2 to 68.7, with the best filtering method (POS alone). The significance of the effect of filtering was reversed when it comes to arguments – with filtering improving scores for *newswire* arguments more than *phrasebook* arguments. SWhen scoring only the arguments attached to a correct predicate and with the correct argument label, *phrasebook* improved from 63.1 to 63.8 and *newswire* improved from 38.1 to 42.2.

All of the tested filtering methods improve precision for identifying predicates compared to the direct projection, but tend to suffer slightly on recall. While the ground-truth fallback method alone had little benefit, it likely enhanced performance more when combined with other filtering techniques. The automatic alignments do not typically incorrectly predict that a word has no alignment in Russian, but the ground-truth method can help in cases where the alignment was incorrect, but another filtering technique removed it (such as when an ARG0 is placed on punctuation), leaving the bilingual dictionary to provide a form of recovery.

Both genres had some amount of ARG0/ARG1 and ARG1/ARG2 confusion, some of which may be due to translation shifts, such as Russian's use of *нравиться*/*'to please'* rather than *'to like'*. For the sake of readability, we collapsed the ARGMs to one column/row in the confusion matrix. Some of our findings regarding ARGMs were that ARGM-LOC was frequently projected in *newswire*, but the Russian PB had no label on those spans. In *phrasebook*, many of the ARGM-MNR projections were labeled as ARGM-ADV in Russian PB, which is a more general purpose modifier.

The only previous work on cross-lingual Russian SRL that we are aware of is the construction of the Universal PropBanks using word alignments, filtering, and bootstrapping to fill in missing SRL labels on the dependents of verbs [2][40]. Although their techniques resulted in sentences with high quality annotations, they are not easily compared with our results here. A significant difference is that they projected only verbal predicates, whereas we include nominalisations and adjectival predicates. Additionally, their method of evaluation was to manually examine only *complete* sentences, where every verb and its dependents received a label.

5.2.5 Error Analysis and Discussion

In this section we present the findings of our error analysis, which covers gaps in the Russian PropBank annotation as well as both systemic and idiosyncratic projection errors.

		Predicted							
		ARG0	ARG1	ARG2	ARG3	ARG4	ARG5	ARGM	Missing
Russian PB	ARG0	**408**	32	2	-	-	-	1	55
	ARG1	31	**414**	23	5	2	-	24	152
	ARG2	5	17	**59**	2	2	-	9	34
	ARG3	1	4	2	**1**	10	-	1	2
	ARG4	-	-	1	-	**4**	-	-	-
	ARGM	3	13	15	1	6	-	**313**	151
	None	124	113	27	3	5	-	168	**3868**

Table 5.4: A confusion matrix showing mislabeling of arguments by the direct projection method on the *phrasebook* data.

		Predicted						
		ARG0	ARG1	ARG2	ARG3	ARG4	ARGM	Missing
Russian PB	ARG0	**44**	13	1	-	-	-	41
	ARG1	4	**65**	3	1	-	7	75
	ARG2	-	5	**14**	-	1	4	16
	ARG3	-	1	-	**-**	-	1	1
	ARG4	-	-	-	-	**1**	-	1
	ARGM	5	4	-	-	-	**56**	66
	None	63	70	10	3	3	104	**5440**

Table 5.5: A confusion matrix showing mislabeling of arguments by the direct projection method on the *newswire* data.

5.2.5.1 Russian PropBank Errors

The English roleset *be.03* describes the auxiliary use of *be*. English annotation guidelines prescribe annotating this predicate with no arguments. Russian has a highly parallel construction with the verb *быть*, but Russian PropBank 2020 marks this usage as the ARGM-MOD of the verb *есть* (*to eat*) in the sentence *"What are we going to eat?"*:

[что ARG1] [мы ARG0] [**будем** ARGM-MOD] [есть PRED] ?

Verbs are sometimes missing from the annotation as well, such as the predicate *сложилось* (*turned out*) in the sentence *"Все сложилось хорошо"* (*"Everything worked out fine"*).

Predicates may be misidentified, such as the interjection *спасибо/thanks*. This particular word is fairly consistently mislabeled as a predicate in Russian PB, and this allows the projection to produce false positives when projecting *thank.01* from English.

There are a number of mistakes and inconsistencies we found related to arguments:

- Sometimes missing ARGM-DIS labels for *да/yes* and *нет/no*
- Inconsistent about marking *должен/obligated* as ARGM-MOD, ARGM-ADV, or not at all.
- *никогда/never* is often marked as ARGM-TMP, whereas it should be ARGM-NEG

Examining some of the ARGMs that the evaluation marked as incorrect highlights some of the challenges inherent in making decisions between them. For instance, English PropBank marks *heart* as the ARGM-EXT of the predicate *love* in the sentence *"I love you with all my **heart**"*. Russian PB chose the more general ARGM-ADV in the equivalent Russian sentence, *"я тебя люблю всем **сердцем**"*.

We find sporadic occurrences of mislabeled arguments as well, such as this case where *вас* (*you*) should be an ARG2 instead of an ARG1 in the sentence (lit. *I you not hear*):

[я $_{\text{ARG0}}$] [вас $_{\text{ARG1}}$] [не $_{\text{ARGM-NEG}}$] [слышу $_{\text{PRED}}$]

5.2.5.2 Systemic Projection Errors

An interesting semantic divergence between Russian and English is the expression of *must*, *should*, *need*, or *have to*. While modal verbs, such as *must* and *should*, in English PropBank are marked as ARGM-MOD arguments to a main verb, these are frequently translated to a construction with *нужно* or *надо* such as in *"We should get a new car"*:

[нам $_{\text{ARG0}}$] [**нужно** $_{\text{ARGM-MOD}}$] [купить $_{\text{купить.01}}$] новую [машину $_{\text{ARG1}}$]

The Russian can be translated more literally as *"To us it is necessary to buy a new car"*. In this case, the projection matches the gold annotation since *should* is an ARGM-MOD in English, but the same Russian word and construction will be mistakenly marked as a predicate, rather than ARGM-MOD, when the source is *need.01*, such as the following example, *“we need ice”*:

[Нам $_{\text{ARG0}}$] [**нужен** $_{\text{need.01}}$] [лед $_{\text{ARG1}}$] .

In Russian PropBank, this sentence does not have a predicate. The adjective *должен* faces similar issues. This word would be literally translated as *obligated*, but is a common way of translating *“need to* X” or *“have to* X”. Russian PB most frequently marks this as an ARGM-MOD, but as with *нужно*, the English translation sometimes uses a predicate, such as *have.02*.

Further complicating matters, *нужен* and *должен* are short-form adjectives. They share common roots with verbs *нуждаться* (*to need*) and *долженствовать* (*to be required to*), respectively, but are not derived from them. Because of this, these are not considered in the current form of Russian PropBank to be predicative. This means that Russian PB will fail to capture any information in a sentence like the previous example of “We need ice”. This raises the question of whether a better representation would be to treat these adjectives as predicates.

Another English modal verb that doesn’t project accurately is *can*. In English, this is marked as an ARGM-MOD, but in Russian PropBank it is a verbal predicate, with the co-occurring verb being its ARG2. In Russian PropBank, there are two predicates in the sentence *“I can call her”*:

[Я $_{\text{ARG1}}$] [**могу** $_{\text{PRED}}$] [позвонить $_{\text{ARG2}}$] ей .

Я могу [**позвонить** $_{\text{PRED}}$] [ей $_{\text{ARG1}}$] .

The one who is can do the action, *я* (*I*), is only attached to the verb *могу* (*can/able*), while the person being called, *ей* (*her*), is attached to the verb *позвонить* (*to call*). This results in a substantial re-arrangement of semantic structure, as can be seen by comparing the two predicate-argument structures annotated by Russian PropBank. Projecting from English PropBank using accurate word alignments results in only a single predicate and *могу* being marked as an ARGM-MOD for it:

[Я $_{\text{ARG0}}$] [могу $_{\text{ARGM-MOD}}$] [**позвонить** $_{\text{PRED}}$] [ей $_{\text{ARG1}}$] .

One type of error that is challenging to remedy arises from nominal predicates in the English PB. The roleset *medicate.01*, as in *"malaria medication"*, is projected in the newswire data to the Russian noun *препарат*. Although the alignment is correct and the translation is perfectly sound, this label is marked as incorrect according to Russian PB. The Russian word *препарат* (*medication*) does not derive from a Russian verb corresponding to the assigned predicate as in English (and is, in fact, a loanword), so would typically not be labeled as a predicate in a Russian SRL corpus. Identifying cases like this, where the projection fails despite perfect alignment and accurate parallel translation, presents difficulties for annotation projection.

5.2.5.3 Idiosyncratic Projection Errors

The word aligner frequently struggles with mapping present tense forms of *"to be"* to the Russian text, which are usually implicit. Rather than omitting these in the alignment, these are often incorrectly aligned to punctuation or particles in the Russian text. The POS filtering we used was designed to remove these spurious alignments.

English newswire texts often use the predicate *say.01* to introduce quotations, such as in

[he $_{\text{ARG0}}$] [said $_{\text{say.01}}$]

The Russian text, rather than the equivalent *сказать* (*to say*), often uses *"по словам ..."* (*"According to the words of..."*) or *"по данным..."* (*"according to the facts..."* as an analogue. These phrases do not lend themselves to a predicate with a *sayer* as its ARG0 as in the English text, so Russian PB does not annotate with a predicate in this situation. Because this translation shift happens frequently, the alignment model often considers the Russian phrase as aligned with *said*, leading to the incorrect projection of *say.01* and its arguments.

5.2.5.4 Future Work

Since the best performance of the word aligners we tested still has an error rate of 15%, there is room for improvement on the alignments that these projection methods rely on. One of the embedding-based alignment methods we tested, AWESOME, used the pre-trained multilingual BERT model for our experiment, but the model allows for fine-tuning on data. Using the parallel LORELEI data to fine-tune it may provide improvements for our target dataset. The word alignments may also be improved by training on additional parallel data, such as ParaCrawl [28], RusLTC [49], or OpenSubtitles [56].

In our method of using a bilingual dictionary as a fallback in the case of the word aligner failing to match a word, we're using the MUSE bilingual dictionary [50]. A shortcoming of this dictionary is that it does not contain all possible forms for translations. For example, the parallel entry for the English past-tense verb *said* only identifies the masculine form of the equivalent past-tense Russian verb *сказал*, and not the feminine or neuter. This approach would benefit from the use of a morphological analysis tool, such as PyMorphy2 [47], to provide additional word forms. Additionally, this particular dictionary is missing some frequently used words, such as the first person singular pronoun *I/я*.

In some cases, we would be better off relying on a bilingual dictionary to choose the appropriate target word than to use the word alignment, but the projection method we used only used a dictionary as a fallback in the case that the word aligner didn't find a match at all. The off-the-shelf word aligner we used, *efmaral*, does not provide insight into the model's confidence in its predictions, but other word aligners have more accessible outputs. Being able to assess the trustworthiness of word alignments could allow us to choose the threshold at which to switch to a different technique, which may be more beneficial than using the word alignment model alone.

When dealing with a small amount of labeled data, but access to a large amount of unlabeled data, a popular approach of weakly supervised learning is bootstrapping, or self-training. Bootstrapping techniques have been shown to improve the quality of projections based on word alignments [2][40][4], by providing a way to overwrite erroneous projections or fill in gaps that were missed by the word alignments.

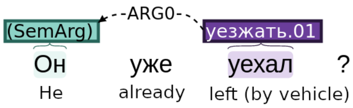
5.3 Russian PropBank 2023

We consider some of the results of the previous Section 5.2 as preliminary due to data quality issues. The differences between the test data and our projections have offered insight into both systemic and idiosyncratic errors. In response, we have done additional annotation and adjudication to improve the quality of the annotations. Particularly, we have been creating new frames [2] and expanding double-annotated and adjudicated coverage of the verbs that were added in the secondary pass, where annotations for predicates without constructed rolesets were added with approximate roles. While the later versions of English PB extensively annotated nominalisations and predicative adjectives, the preliminary pass to add these to the Russian PropBank shows that it is challenging to annotate these consistently and raises complex questions about what is predicative. We have restricted the focus of development to just verbal predicates, which is a decision also made by the Universal PropBanks.

We have added a roleset based on auxiliary *be.03* (*быть.08*). This usage is typically dropped in present tense, but frequently serves to provide tense to imperfective verbs, such as in *мы будем есть* / *we will eat*. This use was previously annotated as an ARGM-MOD for the main verb.

5.3.1 Comparison to English PropBank

English PB may miss semantic distinctions present in Russian. For example, the verb *уезжать/уехать* expresses *leaving by vehicle/animal*, whereas *уйти/уходить* expresses *leaving by one's own power*. The closest English mapping for this predicate, *leave.11*, is divided into two rolesets in Russian: *уезжать.01* and *уйти.01*, with manner is a core argument for *уезжать.01*.



[2] https://github.com/cu-clear/RussianPropbank

Russian uses 23 verbal prefixes that are used to compose new verbs with changes to the aspect or semantics of the verb. English would typically add prepositions or other words to form the equivalent. For example, *пройти* (*to go through*) is formed by the verb *идти* (*to go*) and the prefix *про-* (*through*, or *past*).

We made the decision to only create new rolesets for prefixed verbs if there is a change in meaning beyond an aspectual one or if a prefixed verb requires different core arguments. For example, *пить.01* (*to drink*) can take several prefixes:

- *пить* - to drink (imperfect)
- *выпить* - to drink up / to the end
- *попить* - to drink for a little while / to drink here and there

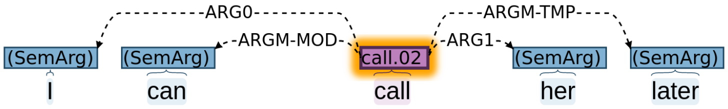
Because these are aspectual differences, these are all aliases to the same roleset. Russian PB may capture less information than English in this case, since English may include ARGMs to provide the additional information.

On the other hand, when a prefix changes core arguments, we construct a new sense. For instance, *сидеть* (*to sit*):

- *сидеть* - to sit
- *просидеть* - to sit through

This latter case almost always requires the specification of a temporal event (*time or event sat through*), and so is defined as an ARG2 role.

A significant difference between English and Russian is the treatment of modals, as we discussed earlier in Section 5.2.5.2. In English, modal verbs (*can*, *may*, *would*, etc.) are uninflected and simply attached to the main predicate as ARGM-MOD.



In Russian, the verb *мочь* can serve a similar role as *can* or *may* and is typically translated as such. In English PropBank, *can* is always treated as an ARGM-MOD and no distinction is made whether the speaker is using it to convey ability or permission, but we add this distinction in Russian PropBank 2023. In the former case, we use the roleset *мочь.01*, which is based on *able.01* and is used for ability/capability and usually agentive and physical:

> Я **могу** бежать очень быстро
> I can (am able to) run really fast

If *мочь* is being used solely to mark modality, conveying permission or possibility, we do not label it as a predicate. This usage is always connected with an ARGM-MOD to the main predicate, if available.

> Вы бы не **смогли** меня подождать?
> Would you be willing to wait for me?

> Он так **может** упасть!
> He might fall like that!

The previous version of Russian PropBank considered both senses to be predicative, using *мочь.03* in the modal sense. We also removed the roleset *давай(те).09* in this update. This usage of the verb *давать* is typically translated as *"let's"*. For example:

> **Давайте** попробуем снова.
> Let's try again.

Because this is functioning to mark the main verb as hortative mood, we consider this to be an ARGM-MOD of the main verb. The English PropBank treats *let* in this sense as a predicate with the roleset *let.01* (*to allow*). The result of this is that there is no difference in the representation compared to *"She let me try again"*.

The use of the ARGM-MOD in English PropBank is restricted to a finite list of modal verbs: Shall, will, should, would, may, might, must, can, and could. Other words that may indicate modality (e.g. *probably*, *possibly*) are often treated as ARGM-ADV, which is used for any adverbial arguments that don't fit into any of the other ARGM categories. In some cases, they may be other types, such as *"I really liked it"* (ARGM-EXT) as opposed to *"It really was true"* (ARGM-ADV).

Our goal with this change to Russian PropBank is to avoid using ARGM-ADV when ARGM-MOD is applicable. This also applies to words such as *возможно* (*possibly*).

5.4 Comparing Universal PropBanks 2.0 With Russian PropBank 2023

We used the Universal PropBanks 2.0 system [40] to project the manual PropBank annotation from the English *LORELEI* sentences to the parallel Russian translations.

5.4.1 Background

As discussed previously in Chapter 2, Universal PropBanks 2.0 consists of automatically generated PropBanks for 23 languages. Similarly to version 1.0, their projection system consists of using a combination of word alignments and filtering techniques to project automatic SRL from English into the target languages, combined with bootstrapping an SRL model to re-label the target sentences and successive re-training on the improved labels. The new 2.0 system uses several improvements compared to their previous version, including utilising more modern syntactic, word alignment, and SRL models. Additionally, they train the re-labeling SRL model jointly on the projected annotations in the target language combined with English gold SRL annotations from *OntoNotes* to improve the performance. They also jointly train on span-based and dependency-based labels, providing output in both forms.

5.4.2 Methods

We provide the gold English PropBank annotations from the *LORELEI* core data to the UPB 2.0 system to produce PropBank annotations on the parallel Russian newswire and phrasebook sentences. The internal bootstrapped Polyglot SRL model is trained on these projected annotations as well as gold English SRL from *OntoNotes*. Since UPB 2.0 does not project participles, which are annotated in Russian PropBank 2023, we do not assess performance on predicates marked as ADJ by the automatic UDPipe parse.

The UPB 2.0 output consists of two forms of labels: dependency heads and span SRL that may include additional arguments without head verification. We will refer to these two versions of projections as **Precise** and **Balanced**, respectively. In order to evaluate against the Russian PB, we determine the dependency heads for the Balanced projections based on the automatic UDPipe dependency parses. If the span contains multiple possible heads, we simply choose one of them, although such a solution is not ideal.

5.4.3 Discussion

The UPB 2.0's SRL re-labeling means that the projection is not limited to using the roleset in the English sentence. This can greatly improve the projection in cases where the parallel sentences are not literal translations.

For example, in one pair of parallel sentences, the English is *"Could you spell that please?"*, but Russian lacks the verb "to spell (something out with letters)". This sentence is translated as *"Вы можете продиктовать по буквам?"* (lit. *"you may dictate by letters?"*). The UPB span projection successfully identifies the Russian verb as the sense dictate.01 and "по буквам" (by letters) as the manner:

[Вы $_{\text{ARGM-MOD}}$] [можете $_{\text{ARGM-MOD}}$] [продиктовать $_{\text{dictate.01}}$] [по буквам $_{\text{ARGM-MNR}}$] .

On the other hand, this can produce an incorrect result where a direct projection would've worked fine. In the sentence *"Is her cell phone not working?"* and equivalent *"Работает ли ее мобильный телефон?"*, the bootstrapped SRL model re-labels the predicate from *work.09* (*function, operate*) as *use.01* (*take advantage of, utilise*). This type of error is difficult to correct, but increased training data for the bootstrapped SRL model may help. It should be noted that this type of error is not detected by our evaluation metric. Because the arguments have the same ordering in both rolesets and we cannot automatically assess whether *use.01* is a reasonable translation of *работать.02*, this mistake escapes notice.

PropBank annotation guidelines dictate that you may not tag multiple core arguments. If an argument is not contiguous, they should be marked using a prefixed argument label, such as an ARG1 with the rest of the argument labeled C-ARG1. We look at the example *"You're just going to make it worse"* (*"Вы все делаете еще хуже"* (lit. "you everything do even worse").

In the Precise projection onto dependency heads, UPB 2.0 produces the following arguments for *make.02* (*cause [to be]*):

[Вы $_{\text{ARG0}}$] [все $_{\text{ARG1}}$] [делаете $_{\text{make.02}}$] еще [хуже $_{\text{ARG1}}$]

The Balanced projection adds yet another ARG1 onto *еще/even*. According to Russian PropBank, the correct annotation uses the делать.03 roleset (*change state*).

[Вы $_{\text{ARG0}}$] [все $_{\text{ARG1}}$] [делаете $_{\text{делать.03}}$] еще [хуже $_{\text{ARG2}}$]

The agent (*Вы/you*) and patient in the projection (*все/everything*) are correct.

A class of error that would be easy to remedy using simple rules is where certain parts of speech are given labels that would never be acceptable in any circumstance, such as on punctuation. Another example is this case where the pronoun *вы* (*you*, formal/plural) is marked as a modal in the sentence *"Could you repeat that?"*:

[Вы $_{\text{ARGM-MOD}}$] [можете $_{\text{ARGM-MOD}}$] [повторить $_{\text{repeat.01}}$] ?

This example is also demonstrative of the fact that certain ARGMs can also be considered closed classes, where any projection onto a token not within that class is rejected. For example, the particles that convey negation (*не, нет, никогда*) are the only ones allowed to be ARGM-NEG.

To summarise, we propose two categories of filtering rules. Based on our understanding of PropBank annotation guidelines and errors made by UPB 2.0, we suggest some rules that can be applied to any language. We also propose some rules that require a basic understanding of the target language. We provide these restrictions for Russian, although they can be similarly adapted for other languages:.

- Language-independent filters
    - ∗ Remove predicates and arguments if the target is a PUNCT or ADP.
    - ∗ PART can only be an ARGM-NEG, ARGM-DIS, ARGM-ADV, or ARGM-MOD
    - ∗ PRON should never have the following ARGM labels: ARGM-MOD, ARGM-NEG, ARGM-MNR, ARGM-EXT, ARGM-ADV, ARGM-ADJ, ARGM-PRR
    - ∗ ARGM-COM can only be a PRON, PROPN, or NOUN.
    - ∗ Only use the most likely target for core arguments, rather than duplicating them. Duplicate core arguments are against PropBank guidelines. In our current methodology, we are prioritising direct children of the predicate; if further criteria are necessary we prioritise NOUN, PROPN, PRON and if there is still ambiguity, we simply choose the first occurrence.
- Language-specific filters
    - ∗ The only token that is allowed to be ARGM-REC are forms of *себя* (*himself, herself, itself, yourself*).
    - ∗ Change ARGM-ADV to ARGM-MOD if it's a form of the following: *Мочь/смочь, можно, бы, должен, нельзя, надо, нужно, возможно.* This list is incomplete, but covers the most frequent cases.
    - ∗ *нет*, *не*, and *никогда* can only be labeled as ARGM-NEG, never as any other argument. No other tokens can be labeled as ARGM-NEG other than these.

5.4.4 Results

We evaluated the Precise and Balanced projections against both Russian PropBank 2020 and 2023, using PriMeSRL [39] for scoring. The performance of the base system and with using the proposed filters as a post-processing step are presented in Tables 5.6 and 5.7 for the *phrasebook* and *newswire* genres, respectively. Since we cannot evaluate whether the sense of a predicate is right, predicates are correct if they are simply in the correct location. Arguments, on the other hand, are only considered correct if they are both correctly placed and labeled. In the case of the 2020 dataset, we evaluate strictly on the gold annotations, not the approximated predicate-argument annotations that were added in a second pass. Because the 2020 dataset has incomplete annotations for some sentences, we only evaluate on the predicates that UPB 2.0 identified and were also present in the gold data, in order not to penalise it for missing gold annotations. For the 2023 dataset, we evaluate on all predicates.

Table 5.6: Performance of UPB 2.0 (Balanced and Precise) with and without the filtering methods on two versions of the Russian PropBank *phrasebook* subset.

(a)

RuPB 2020 Phrasebook			
Balanced			
	P	R	F
Predicates	100.00	76.87	86.92
Arguments	76.51	66.49	71.15
Arguments (filtered)	**78.95 (+2.44)**	**66.75 (+0.26)**	**72.34 (+1.19)**
Precise			
Predicates	100.00	76.87	86.92
Arguments	77.27	66.75	71.63
Arguments (filtered)	**79.19 (+1.92)**	66.75	**72.44 (+0.81)**

(b)

RuPB 2023 Phrasebook			
Balanced			
	P	R	F
Predicates	94.44	90.43	92.39
Arguments	74.91	**74.63**	74.77
Arguments (filtered)	**77.10 (+2.19)**	74.26 (-0.37)	**75.66 (+0.89)**
Precise			
Predicates	94.44	90.43	92.39
Arguments	76.14	73.90	75.00
Arguments (filtered)	**77.61 (+1.47)**	73.90	**75.71 (+0.71)**

The base Balanced projected arguments on the short, simple *phrasebook* sentences attain an F-score of 71.15% and 74.77% on the 2020 and 2023 datasets respectively, with the filtered version gaining 1.19% and 0.89% over this score. As we would expect, the Precise projections benefit less, gaining only 0.81% and 0.71%. The best F-scores are achieved by using filtered Precise projections for *phrasebook*. The base Balanced projections to the more complex *newswire* corpus start at 38.30% and 50.75% and filtering the arguments reduces the 2020 result by a negligible 0.03%, while the 2023 result gains by 0.38%. For this genre, the filtered Balanced projections are the highest scoring.

These filters consistently improve precision (by 0.79-3.60%) across both genres and versions of annotation, but their impact on recall is more variable. On the 2020 dataset, they slightly improve

Table 5.7: Performance of UPB 2.0 (Balanced and Precise) with and without the filtering methods on two versions of the Russian PropBank *newswire* subset.

(a)

RuPB 2020 Newswire			
Balanced			
	P	R	F
Predicates	100.00	57.06	72.66
Arguments	49.69	33.89	40.30
Arguments (filtered)	**53.29** **(+3.60)**	**34.32** **(+0.43)**	**41.75** **(+1.45)**
Precise			
Predicates	100.00	57.06	72.66
Arguments	51.80	**30.38**	**38.30**
Arguments (filtered)	**52.59** **(+0.79)**	30.08 (-0.30)	38.27 (-0.03)

(b)

RuPB 2023 Newswire			
Balanced			
	P	R	F
Predicates	98.48	66.33	79.27
Arguments	72.38	**43.43**	54.29
Arguments (filtered)	**75.76** **(+3.38)**	42.86 (-0.57)	**54.74** **(+0.45)**
Precise			
Predicates	98.48	66.33	79.27
Arguments	73.12	38.86	50.75
Arguments (filtered)	**74.73** **(+1.61)**	38.86	**51.13** **(+0.38)**

recall on the Balanced projections, but slightly detriment it on the 2023 dataset. A contributor to the detriment to recall may partially lay in the filter incorrectly choosing which extraneous core arguments to eliminate. Further analysis may provide better heuristics for determining which argument is most likely to be correct.

Overall, these filters provide a simple and efficient improvement to projection when used as a post-processing step.

5.5 Summary

In this chapter, we described our investigation into using Russian PropBank to provide a test dataset for exploring techniques of developing SRL tools for Russian. We tested multiple word alignment systems on a small manually annotated set of parallel English-Russian sentences. We explored the effects of different filtering techniques on improving the projection from a subset of the gold English *LORELEI* SRL to sentences to their Russian translations, testing against the annotations from the Russian PropBank project.

We identified a need for manual clean-up of the early version of Russian PB and have revised existing frames, expanded annotation to all verbal predicates in a subset of the earlier version with plans to continue, and further developed framing and annotation guidelines.

The latest version of Universal PropBanks 2.0 provides a strong projection framework that is capable of improving recall on the target language compared to previous projection methods through the use of iteratively bootstrapping an SRL model on projected labels. Through our analysis on its performance on Russian PropBank, we have found that many of the errors affecting precision can be identified and corrected using simple heuristics and a basic knowledge of the target language. Although our previous projection method described in Section 5.2 has lower performance than the more recent UPB 2.0, building off the previously tested filtering methods has been fruitful in reliably improving this system's performance across the genres and versions of annotation on Russian PropBank with little computational cost.

Chapter 6

Conclusion

6.1 Active Learning for SRL

In Chapter 3, we presented successful strategies to reduce annotation requirements for developing an SRL model for English [66][67]. We found that using Bayesian Active Learning by Disagreement, as implemented through using dropout during the prediction phase, can reduce the amount of data required to train SRL models compared to both random selection or selecting training instances using the model's output probabilities. We have compared the effect of this in terms of reducing sentences, predicates, or tokens and have discussed the practical considerations in prioritising these metrics.

With the goal of improving over our previous success using BALD, we tested whether selecting individual predicate-argument structures can provide improvements over using all of the predicates in a selected sentence. This more granular approach improves performance on the three narrow domain corpora, but decreases on the larger, more general Ontonotes. Given that our aim is develop semantic resources in new domains, these results support the applicability of our approach to the types of sublanguage corpora we tested.

We have presented several analyses of the sentence and predicate selections over the course of active learning and have found the datasets to vary in terms of rate of vocabulary coverage compared to random selection and the diversity of sentences being chosen. We have also investigated the impact of varying the number of queries per iteration on the learning rate using BALD PREDICATES, which appears sensitive to the size of the test dataset. This has highlighted potential areas for improvement and study, as well as the importance of tailoring active learning to the target data source.

We have found that monitoring the disagreement score over the course of BALD, combined with watching for the performance plateau can potentially be indicative as to how much benefit can be gained by continuing the process. Each corpus varies in terms of how much disagreement is within each batch and how quickly it decreases. While there is no one-fit algorithm for determining the stopping point, these factors may be useful.

We have also tested two methods of providing the most unique training instances in the initial seed data and also limit redundancy: 1) using sentences with a high perplexity according to a language model combined with a simple filter for sentence-level redundancy, and 2) using sentences that offer a diverse coverage according to sentence embeddings. We found neither of these methods beneficial. While similar techniques have previously been effective for WSD and SRL, we speculate that the improvements in SRL models and the word embeddings used as features have made these types of modifications of less benefit compared to the earlier NLP systems.

Overall, our findings demonstrate that active learning remains challenging to apply to new datasets. While many previous active learning methods were simulated and tested for only a single corpus, we have observed many differences between our tested datasets that indicate that prior and future work may not consistently carry over for application to new domains.

In Chapter 4, we investigated whether we can accelerate the training of an AMR parser on corpora with existing SRL annotations. We hypothesised that the overlap of SRL and AMR could also be indicative of an overlap in which training instances are most informative. By simulating active learning on the SRL annotations, we can obtain an ordering of the priority of these instances. However, we find that the SRL AL instances are less useful to training an AMR model than random sentences, likely due to factors such as AL for SRL not being interested in sentences without explicit predicates and the extent to which an AMR parser must learn how to decompose concepts, which is not required for SRL.

In Chapter 5, we compared various word alignment models for the use of projecting SRL annotations from English to Russian. Our evaluation and error analysis of this on Russian PropBank led to several unsupervised modifications for filtering that improve precision, as well as improve recall through using a bilingual dictionary. These experiments led to additional expansion and refinement of Russian PropBank, which is ongoing.

We then examined using the Universal PropBanks 2.0 system to project SRL from English to Russian and evaluating on both the previous 2020 and latest 2023 versions of Russian PropBank. We identified several areas of improvements based on a combination of simple language-specific rules and language-independent filtering for standard PropBank annotation. These filtering methods, applied as post-processing, provide a consistent improvement to the system.

6.2 Future Directions

Although our work has demonstrated the significant benefits to using active learning for SRL, many open questions remain. Our experiments were limited to one model architecture and it would be valuable to test the robustness of these strategies on differing architectures. Our analysis of the selections being made point towards promising avenues of tuning the algorithm further by enforcing more diversity in the batches of selected queries. We've shed light on some of the variance that may be encountered when applying active learning to different corpora, but there is more work to be done on understanding the relevant factors in order to most effectively choose the selection strategy and parameters for real-world use.

Future work in applying active learning to AMRs could examine using the model's output probabilities to select training instances, although the prior literature and our results on SRL indicate that this would be less beneficial than using a strategy such as BALD. Using BALD in this context would require aligning graph structures to produce agreement scores, which may be feasible through existing alignment algorithms (such as SMATCH) or new innovations.

Although we have focused on Russian as our target language, Universal PropBanks 2.0 has been evaluated on manual gold annotations (using English PropBank rolesets) for Polish, Vietnamese, Portuguese, and French. The language-independent filters can be universally applied, but native speakers could supply the lists of words that express negation, modality, and reciprocals in these languages in order to apply the language-specific filters as well. Additionally, the proposed filtering rules for UPB 2.0 may further enhance performance as a step before bootstrapping, rather than as post-processing.

Ultimately, we want to understand whether these projected annotations that are innately biased towards English are sufficient for input into downstream applications compared to designing and annotating semantic role labeling frames for the target language. To this end, we must improve the automatic projections to a point where a fair comparison can be made by testing the results of both strategies on another task as input.

The recent innovations and availability of large language models offer exciting new avenues to reach our goal of expanding semantic machine understanding to new areas. Leveraging these new technologies' aggregation of large amounts of data and generative capabilities may further improve the approaches we have used (such as by using them for generating word alignments), and conversely, applications of LLMs may be benefited by the intermediary use of symbolic representations such as SRL and AMR in order to provide distillation of information, constraints, and explainability. The integration of architecture innovations, large-scale training, linguistic reasoning, and symbolic representations collectively can bring about useful, intelligent systems to both synthesise information and provide a means of understanding their reliability and trustworthiness.

Bibliography

[1] Koray AK and Olcay Taner Yıldız. Automatic Propbank generation for Turkish. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), pages 33–41, Varna, Bulgaria, September 2019. INCOMA Ltd.

[2] Alan Akbik, Laura Chiticariu, Marina Danilevsky, Yunyao Li, Shivakumar Vaithyanathan, and Huaiyu Zhu. Generating high quality Proposition Banks for multilingual semantic role labeling. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 397–407, Beijing, China, July 2015. Association for Computational Linguistics.

[3] Daniel Albright, Arrick Lanfranchi, Anwen Fredriksen, IV Styler, William F, Colin Warner, Jena D Hwang, Jinho D Choi, Dmitriy Dligach, Rodney D Nielsen, James Martin, Wayne Ward, Martha Palmer, and Guergana K Savova. Towards comprehensive syntactic and semantic annotations of the clinical narrative. Journal of the American Medical Informatics Association, 20(5):922–930, 01 2013.

[4] Maryam Aminian, Mohammad Sadegh Rasooli, and Mona Diab. Transferring semantic roles using translation and syntactic information. In Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 13–19, Taipei, Taiwan, November 2017. Asian Federation of Natural Language Processing.

[5] Maryam Aminian, Mohammad Sadegh Rasooli, and Mona Diab. Cross-lingual transfer of semantic roles: From raw text to semantic roles. In Proceedings of the 13th International Conference on Computational Semantics - Long Papers, pages 200–210, Gothenburg, Sweden, May 2019. Association for Computational Linguistics.

[6] Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 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, August 1998. Association for Computational Linguistics.

[7] Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. Abstract Meaning Representation for sembanking. In Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, pages 178–186, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[8] Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. One spring to rule them both: Symmetric amr semantic parsing and generation without a complex pipeline. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 35, pages 12564–12573, 2021.

[9] Rajesh Bhatt, Bhuvana Narasimhan, Martha Palmer, Owen Rambow, Dipti Sharma, and Fei Xia. A multi-representational and multi-layered treebank for Hindi/Urdu. In Proceedings of the Third Linguistic Annotation Workshop (LAW III), pages 186–189, Suntec, Singapore, August 2009. Association for Computational Linguistics.

[10] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. In Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37, ICML'15, page 1613–1622. JMLR.org, 2015.

[11] Alexey Borisov, Jacob Dlougach, and Irina Galinskaya. Yandex school of data analysis machine translation systems for WMT13. In Proceedings of the Eighth Workshop on Statistical Machine Translation, pages 99–103, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[12] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. Computational Linguistics, 19(2):263–311, 1993.

[13] Rui Cai and Mirella Lapata. Alignment-free cross-lingual semantic role labeling. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 3883–3894, Online, November 2020. Association for Computational Linguistics.

[14] Shu Cai and Kevin Knight. Smatch: an evaluation metric for semantic feature structures. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 748–752, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[15] Aditi Chaudhary, Antonios Anastasopoulos, Zaid Sheikh, and Graham Neubig. Reducing confusion in active learning for part-of-speech tagging. Transactions of the Association for Computational Linguistics, 9:1–16, 2021.

[16] Liang Chen, Peiyi Wang, Runxin Xu, Tianyu Liu, Zhifang Sui, and Baobao Chang. ATP: AMRize then parse! enhancing AMR parsing with PseudoAMRs. In Findings of the Association for Computational Linguistics: NAACL 2022, pages 2482–2496, Seattle, United States, July 2022. Association for Computational Linguistics.

[17] Hyonsu Choe, Jiyoon Han, Hyejin Park, Tae Hwan Oh, and Hansaem Kim. Building Korean Abstract Meaning Representation corpus. In Proceedings of the Second International Workshop on Designing Meaning Representations, pages 21–29, Barcelona Spain (online), December 2020. Association for Computational Linguistics.

[18] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In Proceedings of the 58th Annual

Meeting of the Association for Computational Linguistics, pages 8440–8451, Online, July 2020. Association for Computational Linguistics.

[19] Angel Daza and Anette Frank. X-SRL: A parallel cross-lingual semantic role labeling dataset. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 3904–3914, Online, November 2020. Association for Computational Linguistics.

[20] Dmitriy Dligach and Martha Palmer. Good seed makes a good crop: Accelerating active learning using language modeling. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 6–10, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[21] Li Dong, Chris Quirk, and Mirella Lapata. Confidence modeling for neural semantic parsing. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 743–753, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[22] Zi-Yi Dou and Graham Neubig. Word alignment by fine-tuning embeddings on parallel corpora. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 2112–2128, Online, April 2021. Association for Computational Linguistics.

[23] David Dowty. Thematic proto-roles and argument selection. language, 67(3):547–619, 1991.

[24] Tomasz Dryjański, Monika Zaleska, Bartek Kuźma, Artur Błażejewski, Zuzanna Bordzicka, Paweł Bujnowski, Klaudia Firlag, Christian Goltz, Maciej Grabowski, Jakub Jończyk, Grzegorz Kłosiński, Bartłomiej Paziewski, Natalia Paszkiewicz, Jarosław Piersa, and Piotr Andruszkiewicz. Samsung research Poland (SRPOL) at SemEval-2022 task 9: Hybrid question answering using semantic roles. In Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 1263–1273, Seattle, United States, July 2022. Association for Computational Linguistics.

[25] R. Duerr, A. Thessen, C. J. Jenkins, M. Palmer, S. Myers, and S. Ramdeen. The ClearEarth Project: Preliminary Findings from Experiments in Applying the CLEARTK NLP Pipeline and Annotation Tools Developed for Biomedicine to the Earth Sciences. In AGU Fall Meeting Abstracts, volume 2016, pages IN11B–1625, December 2016.

[26] Chris Dyer, Victor Chahuneau, and Noah A. Smith. A simple, fast, and effective reparameterization of IBM model 2. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 644–648, Atlanta, Georgia, June 2013. Association for Computational Linguistics.

[27] Maud Ehrmann, Marco Turchi, and Ralf Steinberger. Building a multilingual named entity-annotated corpus using annotation projection. In Proceedings of the International Conference Recent Advances in Natural Language Processing 2011, pages 118–124, Hissar, Bulgaria, September 2011. Association for Computational Linguistics.

[28] Miquel Esplà, Mikel Forcada, Gema Ramírez-Sánchez, and Hieu Hoang. ParaCrawl: Web-scale parallel corpora for the languages of the EU. In Proceedings of Machine Translation

Summit XVII Volume 2: Translator, Project and User Tracks, pages 118–119, Dublin, Ireland, August 2019. European Association for Machine Translation.

[29] Hao Fei, Meishan Zhang, and Donghong Ji. Cross-lingual semantic role labeling with high-quality translated training corpus. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7014–7026, Online, July 2020. Association for Computational Linguistics.

[30] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48, ICML'16, page 1050–1059. JMLR.org, 2016.

[31] Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. AllenNLP: A deep semantic natural language processing platform. In Proceedings of Workshop for NLP Open Source Software (NLP-OSS), pages 1–6, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[32] Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. Deep semantic role labeling: What works and what's next. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 473–483, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[33] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. arXiv preprint arXiv:1112.5745, 2011.

[34] Rebecca Hwa, Philip Resnik, Amy Weinberg, Clara Cabezas, and Okan Kolak. Bootstrapping parsers via syntactic projection across parallel texts. Nat. Lang. Eng., 11(3):311–325, September 2005.

[35] Fariz Ikhwantri, Samuel Louvan, Kemal Kurniawan, Bagas Abisena, Valdi Rachman, Alfan Farizki Wicaksono, and Rahmad Mahendra. Multi-task active learning for neural semantic role labeling on low resource conversational corpus. In Proceedings of the Workshop on Deep Learning Approaches for Low-Resource NLP, pages 43–50, Melbourne, July 2018. Association for Computational Linguistics.

[36] Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1627–1643, Online, November 2020. Association for Computational Linguistics.

[37] Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1627–1643, Online, November 2020. Association for Computational Linguistics.

[38] Ishan Jindal, Yunyao Li, Siddhartha Brahma, and Huaiyu Zhu. CLAR: A cross-lingual argument regularizer for semantic role labeling. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 3113–3125, Online, November 2020. Association for Computational Linguistics.

[39] Ishan Jindal, Alexandre Rademaker, Khoi-Nguyen Tran, Huaiyu Zhu, Hiroshi Kanayama, Marina Danilevsky, and Yunyao Li. PriMeSRL-eval: A practical quality metric for semantic role labeling systems evaluation. In Findings of the Association for Computational Linguistics: EACL 2023, pages 1806–1818, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

[40] Ishan Jindal, Alexandre Rademaker, Michał Ulewicz, Ha Linh, Huyen Nguyen, Khoi-Nguyen Tran, Huaiyu Zhu, and Yunyao Li. Universal Proposition Bank 2.0. In Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 1700–1711, Marseille, France, June 2022. European Language Resources Association.

[41] Pavan Kapanipathi, Ibrahim Abdelaziz, Srinivas Ravishankar, Salim Roukos, Alexander Gray, Ramón Fernandez Astudillo, Maria Chang, Cristina Cornelio, Saswati Dana, Achille Fokoue, Dinesh Garg, Alfio Gliozzo, Sairam Gurajada, Hima Karanam, Naweed Khan, Dinesh Khandelwal, Young-Suk Lee, Yunyao Li, Francois Luus, Ndivhuwo Makondo, Nandana Mihindukulasooriya, Tahira Naseem, Sumit Neelam, Lucian Popa, Revanth Gangi Reddy, Ryan Riegel, Gaetano Rossiello, Udit Sharma, G P Shrivatsa Bhargav, and Mo Yu. Leveraging Abstract Meaning Representation for knowledge base question answering. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 3884–3894, Online, August 2021. Association for Computational Linguistics.

[42] Richard Kittredge. Sublanguages. American Journal of Computational Linguistics, 8(2):79–84, 1982.

[43] Natalia Klyueva and Ondřej Bojar. UMC 0.1: Czech-Russian-English Multilingual Corpus. In Proceedings of International Conference Corpus Linguistics, pages 188–195, 2008.

[44] Kevin Knight, Bianca Badarau, Laura Baranescu, Claire Bonial, Madalina Bardocz, Kira Griffitt, Ulf Hermjakob, Daniel Marcu, Martha Palmer, Tim O'Gorman, et al. Abstract meaning representation (amr) annotation release 3.0. 2021.

[45] Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In MT summit, volume 5, pages 79–86. Citeseer, 2005.

[46] David Kolovratnık, Natalia Klyueva, and Ondrej Bojar. Statistical machine translation between related and unrelated languages. In Proceedings of the Conference on Theory and Practice on Information Technologies, pages 31–36. Citeseer, 2009.

[47] Mikhail Korobov. Morphological analyzer and generator for Russian and Ukrainian languages. In Mikhail Yu. Khachay, Natalia Konstantinova, Alexander Panchenko, Dmitry Ignatov, and Valeri G. Labunets, editors, Analysis of Images, Social Networks and Texts, pages 320–332, Cham, 2015. Springer International Publishing.

[48] Omri Koshorek, Gabriel Stanovsky, Yichu Zhou, Vivek Srikumar, and Jonathan Berant. On the limits of learning to actively learn semantic representations. In Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL), pages 452–462, Hong Kong, China, November 2019. Association for Computational Linguistics.

[49] Andrey Kutuzov and Maria Kunilovskaya. Russian learner translator corpus. In Petr Sojka, Aleš Horák, Ivan Kopeček, and Karel Pala, editors, Text, Speech and Dialogue, pages 315–323, Cham, 2014. Springer International Publishing.

[50] Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data. In International Conference on Learning Representations, 2018.

[51] Bin Li, Yuan Wen, Li Song, Weiguang Qu, and Nianwen Xue. Building a Chinese AMR bank with concept and relation alignments. Linguistic Issues in Language Technology, 18, July 2019.

[52] Changmao Li and Jeffrey Flanigan. Improving neural machine translation with the Abstract Meaning Representation by combining graph and sequence transformers. In Proceedings of the 2nd Workshop on Deep Learning on Graphs for Natural Language Processing (DLG4NLP 2022), pages 12–21, Seattle, Washington, July 2022. Association for Computational Linguistics.

[53] Shoushan Li, Yunxia Xue, Zhongqing Wang, and Guodong Zhou. Active learning for cross-domain sentiment classification. In Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, IJCAI '13, page 2127–2133. AAAI Press, 2013.

[54] Percy Liang, Ben Taskar, and Dan Klein. Alignment by agreement. In Proceedings of the Human Language Technology Conference of the NAACL, Main Conference, pages 104–111, New York City, USA, June 2006. Association for Computational Linguistics.

[55] Ha Linh and Huyen Nguyen. A case study on meaning representation for Vietnamese. In Proceedings of the First International Workshop on Designing Meaning Representations, pages 148–153, Florence, Italy, August 2019. Association for Computational Linguistics.

[56] Pierre Lison and Jörg Tiedemann. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), Paris, France, may 2016. European Language Resources Association (ELRA).

[57] Fei Liu, Jeffrey Flanigan, Sam Thomson, Norman Sadeh, and Noah A. Smith. Toward abstractive summarization using semantic representations. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1077–1086, Denver, Colorado, May–June 2015. Association for Computational Linguistics.

[58] Olga Lyashevskaya and Egor Kashkin. Framebank: A database of russian lexical constructions. In Mikhail Yu. Khachay, Natalia Konstantinova, Alexander Panchenko, Dmitry Ignatov, and Valeri G. Labunets, editors, Analysis of Images, Social Networks and Texts, pages 350–360, Cham, 2015. Springer International Publishing.

[59] Sean MacAvaney, Arman Cohan, and Nazli Goharian. GUIR at SemEval-2017 task 12: A framework for cross-domain clinical temporal information extraction. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), pages 1024–1029, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[60] Diego Marcheggiani, Jasmijn Bastings, and Ivan Titov. Exploiting semantics in neural machine translation with graph convolutional networks. In Proceedings of the 2018 Conference of the

North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 486–492, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

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

[62] Ritwik Mishra and Tirthankar Gayen. Automatic lossless-summarization of news articles with abstract meaning representation. Procedia Computer Science, 135:178–185, 2018.

[63] Sarah Moeller, Irina Wagner, Martha Palmer, Kathryn Conger, and Skatje Myers. The Russian PropBank. In Proceedings of the 12th Language Resources and Evaluation Conference, pages 5995–6002, Marseille, France, May 2020. European Language Resources Association.

[64] Muhidin Mohamed and Mourad Oussalah. Srl-esa-textsum: A text summarization approach based on semantic role labeling and explicit semantic analysis. Information Processing & Management, 56(4):1356–1372, 2019.

[65] Paola Monachesi, Gerwert Stevens, and Jantine Trapman. Adding semantic role annotation to a corpus of written Dutch. In Proceedings of the Linguistic Annotation Workshop, pages 77–84, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[66] Skatje Myers and Martha Palmer. Tuning deep active learning for semantic role labeling. In Proceedings of the 14th International Conference on Computational Semantics (IWCS 2021, forthcoming), 2021.

[67] Skatje Myers and Martha Palmer. Leveraging active learning to minimise SRL annotation across corpora. In Proceedings of the The 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023), pages 399–408, Toronto, Canada, July 2023. Association for Computational Linguistics.

[68] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. Computational Linguistics, 29(1):19–51, 2003.

[69] Tim O'Gorman, Sameer Pradhan, Martha Palmer, Julia Bonn, Katie Conger, and James Gung. The new Propbank: Aligning Propbank with AMR through POS unification. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

[70] Elif Oral, Ali Acar, and Gülşen Eryiğit. Abstract meaning representation of turkish. Natural Language Engineering, page 1–30, 2022.

[71] Robert Östling and Jörg Tiedemann. Efficient word alignment with Markov Chain Monte Carlo. Prague Bulletin of Mathematical Linguistics, 106:125–146, October 2016.

[72] Sebastian Padó and Mirella Lapata. Cross-lingual annotation projection of semantic roles. J. Artif. Int. Res., 36(1):307–340, September 2009.

[73] Martha Palmer, Olga Babko-Malaya, Ann Bies, Mona Diab, Mohamed Maamouri, Aous Mansouri, and Wajdi Zaghouani. A pilot Arabic Propbank. In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), Marrakech, Morocco, May 2008. European Language Resources Association (ELRA).

[74] Martha Palmer, Daniel Gildea, and Paul Kingsbury. The Proposition Bank: An annotated corpus of semantic roles. Computational Linguistics, 31(1):71–106, 2005.

[75] Daniel Peterson, Martha Palmer, and Shumin Wu. Focusing annotation for semantic role labeling. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA).

[76] Sameer Pradhan, Julia Bonn, Skatje Myers, Kathryn Conger, Tim O'gorman, James Gung, Kristin Wright-bettner, and Martha Palmer. PropBank comes of Age—Larger, smarter, and more diverse. In Proceedings of the 11th Joint Conference on Lexical and Computational Semantics, pages 278–288, Seattle, Washington, July 2022. Association for Computational Linguistics.

[77] Sameer Pradhan, Wayne Ward, and James Martin. Towards robust semantic role labeling. In Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, pages 556–563, Rochester, New York, April 2007. Association for Computational Linguistics.

[78] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[79] Reinhard Rapp. Using semantic role labeling to improve neural machine translation. In Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 3079–3083, Marseille, France, June 2022. European Language Resources Association.

[80] Greg Schohn and David Cohn. Less is more: Active learning with support vector machines. In Proceedings of the Seventeenth International Conference on Machine Learning, ICML '00, page 839–846, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[81] Karin Kipper Schuler. VerbNet: A broad-coverage, comprehensive verb lexicon. University of Pennsylvania, 2005.

[82] Hinrich Schütze, Emre Velipasaoglu, and Jan O Pedersen. Performance thresholding in practical text classification. In Proceedings of the 15th ACM international conference on Information and knowledge management, pages 662–671, 2006.

[83] Burr Settles. Active learning literature survey. 2009.

[84] Yanyao Shen, Hyokun Yun, Zachary Lipton, Yakov Kronrod, and Animashree Anandkumar. Deep active learning for named entity recognition. In Proceedings of the 2nd Workshop on Representation Learning for NLP, pages 252–256, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[85] Yanyao Shen, Hyokun Yun, Zachary Lipton, Yakov Kronrod, and Animashree Anandkumar. Deep active learning for named entity recognition. In Proceedings of the 2nd Workshop on Representation Learning for NLP, pages 252–256, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[86] Janaki Sheth, Young-Suk Lee, Ramón Fernandez Astudillo, Tahira Naseem, Radu Florian, Salim Roukos, and Todd Ward. Bootstrapping multilingual AMR with contextual word alignments. In Proceedings of the 16th Conference of the European Chapter of the Association for

Computational Linguistics: Main Volume, pages 394–404, Online, April 2021. Association for Computational Linguistics.

[87] Peng Shi and Jimmy Lin. Simple BERT models for relation extraction and semantic role labeling. arXiv preprint arXiv:1904.05255, 2019.

[88] Aditya Siddhant and Zachary C. Lipton. Deep Bayesian active learning for natural language processing: Results of a large-scale empirical study. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2904–2909, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[89] Ard Snijders, Douwe Kiela, and Katerina Margatina. Investigating multi-source active learning for natural language inference. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 2187–2209, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

[90] Marco Antonio Sobrevilla Cabezudo and Thiago Pardo. Towards a general Abstract Meaning Representation corpus for Brazilian Portuguese. In Proceedings of the 13th Linguistic Annotation Workshop, pages 236–244, Florence, Italy, August 2019. Association for Computational Linguistics.

[91] Hye-Jeong Song, Chan-Young Park, Jung-Kuk Lee, Min-Ji Lee, Yoon-Jeong Lee, Jong-Dae Kim, and Yu-Seop Kim. Construction of Korean semantic annotated corpus. In Tai-hoon Kim, Jianhua Ma, Wai-chi Fang, Yanchun Zhang, and Alfredo Cuzzocrea, editors, Computer Applications for Database, Education, and Ubiquitous Computing, pages 265–271, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[92] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: Masked and permuted pre-training for language understanding. arXiv preprint arXiv:2004.09297, 2020.

[93] Linfeng Song, Daniel Gildea, Yue Zhang, Zhiguo Wang, and Jinsong Su. Semantic neural machine translation using AMR. Transactions of the Association for Computational Linguistics, 7:19–31, 2019.

[94] Milan Straka and Jana Straková. Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. In Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, pages 88–99, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[95] Oscar Täckström, Ryan McDonald, and Jakob Uszkoreit. Cross-lingual word clusters for direct transfer of linguistic structure. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 477–487, Montréal, Canada, June 2012. Association for Computational Linguistics.

[96] Jörg Tiedemann. Rediscovering annotation projection for cross-lingual parser induction. In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1854–1864, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics.

[97] Jörg Tiedemann. Parallel data, tools and interfaces in opus. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Ugur Dogan, Bente Maegaard, Joseph

Mariani, Jan Odijk, and Stelios Piperidis, editors, Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey, may 2012. European Language Resources Association (ELRA).

[98] Katrin Tomanek, Florian Laws, Udo Hahn, and Hinrich Schütze. On proper unit selection in active learning: Co-selection effects for named entity recognition. In Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing, pages 9–17, Boulder, Colorado, June 2009. Association for Computational Linguistics.

[99] Lonneke van der Plas, Marianna Apidianaki, and Chenhua Chen. Global methods for cross-lingual semantic role and predicate labelling. In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1279–1290, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics.

[100] Lonneke van der Plas, Paola Merlo, and James Henderson. Scaling up automatic cross-lingual semantic role annotation. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 299–304, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[101] Chenguang Wang, Laura Chiticariu, and Yunyao Li. Active learning for black-box semantic role labeling with neural factors. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17, pages 2908–2914, 2017.

[102] Shira Wein, Lucia Donatelli, Ethan Ricker, Calvin Engstrom, Alex Nelson, Leonie Harter, and Nathan Schneider. Spanish Abstract Meaning Representation: Annotation of a general corpus. In Northern European Journal of Language Technology, Volume 8, Copenhagen, Denmark, 2022. Northern European Association of Language Technology.

[103] Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston. OntoNotes Release 5.0. 2013.

[104] Nianwen Xue and Martha Palmer. Annotating the propositions in the Penn Chinese treebank. In Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, pages 47–54, Sapporo, Japan, July 2003. Association for Computational Linguistics.

[105] David Yarowsky and Grace Ngai. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In Second Meeting of the North American Chapter of the Association for Computational Linguistics, 2001.

[106] Wen-tau Yih, Matthew Richardson, Chris Meek, Ming-Wei Chang, and Jina Suh. The value of semantic parse labeling for knowledge base question answering. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 201–206, Berlin, Germany, August 2016. Association for Computational Linguistics.

[107] Hongming Zhang, Haoyu Wang, and Dan Roth. Zero-shot Label-aware Event Trigger and Argument Classification. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 1331–1340, Online, August 2021. Association for Computational Linguistics.

[108] Zhisong Zhang, Emma Strubell, and Eduard Hovy. A survey of active learning for natural language processing. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 6166–6190, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[109] Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. Semantics-aware bert for language understanding. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, pages 9628–9635, 2020.

[110] Jingbo Zhu and Eduard Hovy. Active learning for word sense disambiguation with methods for addressing the class imbalance problem. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pages 783–790, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[111] Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. The United Nations parallel corpus v1.0. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pages 3530–3534, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA).