Tuning Deep Active Learning for Semantic Role Labeling

Skatje Myers
University of Colorado at Boulder
skatje.myers@colorado.edu

Martha Palmer
University of Colorado at Boulder
mpalmer@colorado.edu

Abstract

Active learning has been shown to reduce annotation requirements for numerous natural language processing tasks, including semantic role labeling (SRL). SRL involves labeling argument spans for potentially multiple predicates in a sentence, which makes it challenging to aggregate the numerous decisions into a single score for determining new instances to annotate. In this paper, we apply two ways of aggregating scores across multiple predicates in order to choose query sentences with two methods of estimating model certainty: using the neural network's outputs and using dropout-based Bayesian Active Learning by Disagreement. We compare these methods with three passive baselines — random sentence selection, random whole-document selection, and selecting sentences with the most predicates — and analyse the effect these strategies have on the learning curve with respect to reducing the number of annotated sentences and predicates to achieve high performance.

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 (MacAvaney et al., 2017) and question-answering systems (Yih et al., 2016). 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 languages such as English (Weischedel et al., 2013), but such resources are lacking in most other languages. Additionally, those corpora created may still not translate well to other in-language domains, due to sentence structure or domain-specific vocabulary. Creation of additional annotated corpora requires a significant amount of time and often the hiring of domain experts, causing a bottleneck for developing advanced NLP tools for other languages and domains.

Active learning (AL) focuses on choosing only the most informative and least repetitive instances to have annotated, thereby reducing the total needed annotation to train a supervised model, without sacrificing performance. This is done by iteratively re-training the model and assessing its confidence in its predictions in order to choose additional data for annotation that would have maximal impact on the learning rate.

Traditionally, 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. In lieu of this, Gal and Ghahramani (2016) found that we can estimate model confidence by calculating the rate of disagreement of multiple Monte Carlo draws from a stochastic model, accomplished by utilising dropout during forward passes. Previous work (Siddhant and Lipton, 2018)(Shen et al., 2017) has combined this finding with Bayesian Active Learning by Disagreement (Houlsby et al., 2011) as a way of selecting informative instances for active learning for SRL and other NLP tasks; hereafter referred to as DO-BALD.

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 work, we compare two ways of aggregating confidence

Proceedings of the 14th International Conference on Computational Semantics, pages 212–221
June 17–18, 2021. ©2021 Association for Computational Linguistics

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.

We will discuss the practical workflow of SRL annotation and the way this must be considered in utilising active learning effectively to create new datasets. Although the current standard data selection methodology for SRL corpora, which typically involves selecting entire documents, leaves much room for improvement by even passive strategies, we will show that active learning can provide significant reductions in annotation of both number of sentences and number of predicates. We aim to provide this comparison within the broader context and understanding of SRL annotation in practice.

2 Background

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

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

Table 1: PropBank roleset for *give.01*.

Proposition Bank (PropBank) (Palmer et al., 2005) is verb-oriented semantic representation. Predicates in text are assigned a *roleset ID* based on the sense of the word, such as play.01 (*to play a game*) or play.02 (*to play a role*). The roleset determines the permissible semantic roles, or arguments, for that predicate. The core arguments are given generalised numbered labels, ARG0 through ARG5. Typically an ARG0 is the agent or experiencer, while ARG1 is typically the patient or theme of the predicate. Additionally, there are modifier arguments to incorporate other semantically relevant information such as location (ARGM-LOC) and direction (ARGM-DIR). The following is an example of the arguments related to the predicate "give" according to the roleset in Table 1:

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

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

Many large corpora have been annotated in English, such as Ontonotes (Weischedel et al., 2013). 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 multiword expressions (O'Gorman et al., 2018), 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. Additionally, there are many domain-specific SRL corpora, such as clinical records (Albright et al., 2013) and the geosciences (Duerr et al., 2016). 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.

Much of the text annotated with PropBank annotations was annotated using Jubilee (Choi et al., 2010). The text is set up to be presented to annotators in the order of the predicate's lemma, enabling annotators to concentrate on the differences between rolesets of particular lemmas and providing efficiency through minimising context-switching. With this methodology, annotation time can primarily be reduced by minimising the number of predicates being annotated.

While this setup is typical of large-scale annotation projects, it's less feasible in the context of active learning. If each iteration results in querying annotators for only 100 sentences, there is little benefit to splitting annotation tasks based on lemmas. The more practical approach is to annotate on a sentence-by-sentence basis. In this case, reducing predicates is still beneficial, but since the cognitive burden of reading and understand the sentence must be done anyway, reducing the number of sentences is of high importance.

When new datasets are annotated, typically entire documents are chosen. Annotation projects frequently do several layers of annotation on the same text, which may include NER, syntactic parsing, SRL, coreference resolution, and event coreference. In the case of SRL, this results in numerous sentences with the same topic and vocabulary being used. The random selection of sentences used as a baseline in active learning studies may be an improvement over the selection criteria used in practice since the distribution of it will result in a more diverse dataset. For this reason, it's important when discussing how much annotation reduction an AL technique provides by selecting individual sentences to compare to the learning curve of random selection, rather than the full dataset. Our experiments include a whole-document selection method to provide comparison.

3 Related Work

Active learning has been utilised with success in numerous NLP tasks, such as named entity recognition (Shen et al., 2017), word sense disambiguation (Zhu and Hovy, 2007), and sentiment classification (Li et al., 2013). In recent years, active learning has been applied to SRL. Since probabilities from off-the-shelf NN models may sometimes be inaccessible, Wang et al. (2017) 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. (2019) 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, 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.

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 (Ikhwantri et al., 2018), 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.

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 (2016) 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 (2018), including SRL. For their SRL experiments, they used a neural SRL model based on the He et al. (2017) 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 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. This selection strategy is similar to the selection method we propose in this paper, but with several differences. The most significant difference is that the authors treat agreement between predictions as all-or-nothing, rather than allowing partial agreement based on arguments. They also are using a higher number of predictions (100 per sentence as opposed to 5 per predicate) to calculate disagreement between, which may be necessary in this all-or-nothing approach. In contrast, we consider each predicate-argument label sequence independently.

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.

4 Data

We used two independent datasets for our experiments: The English section of Ontonotes (version 5.0) (Weischedel et al., 2013) with the latest frame updates (O'Gorman et al., 2018) and the colon cancer portion of THYME (Albright et al., 2013).

Ontonotes 5.0 consists of 1.5 million words across multiple genres. The majority of this data is sourced from news, but it also includes telephone conversations, text from The Bible, and web data. THYME is comprised of clinical notes and pathology reports of colon and brain cancer patients. For our experiments, we used only the colon cancer portion. The data is split into training, validation, and test subsets.

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 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.

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

5 Model

We used AllenNLP's (Gardner et al., 2018) implementation of a state-of-the-art BERT-based model (Shi and Lin, 2019). 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 batches of 100 sentences re-trained from scratch. In the case of the whole-document baseline, for the creation of each batch, we selected random documents until the number of sentences selected met or exceeded 100.

6 Selection Methods

6.1 Model Output

We used 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 for each token and then normalised by sentence length. This results in a single confidence score for the label sequence.

6.2 DO-BALD

The model output of neural networks are a poor estimate of confidence, due to their nonlinearity and tendency to overfit and be overconfident in their predictions (Gal and Ghahramani, 2016)(Dong et al., 2018).

Using Monte Carlo dropout as a Bayesian approximation of uncertainty, as proposed by Gal and Ghahramani (2016), 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 (2016) used between 1000 and 10 forward passes in their experiments and Siddhant and Lipton (2018) 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.

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

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 2, 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.

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 (2018)'s 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.

6.3 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 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 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*).

[ARG0 The governor] [ARGM-OutputD could] [ARGM-NEG n't] [Pred make it], so the lieutenant governor came instead.

The governor could n't make it, so [ARG1 the lieutenant governor] [Pred 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.

6.4 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.

7 Results

Out results are reported as a learning curve across number of sentences (Figures 1, 3) and predicates (Figures 2, 4) present in the training pool after each iteration. Selected F-scores for the methods are reported according to number of sentences (Table 3) and approximate number of predicates (Table 4) 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: 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.

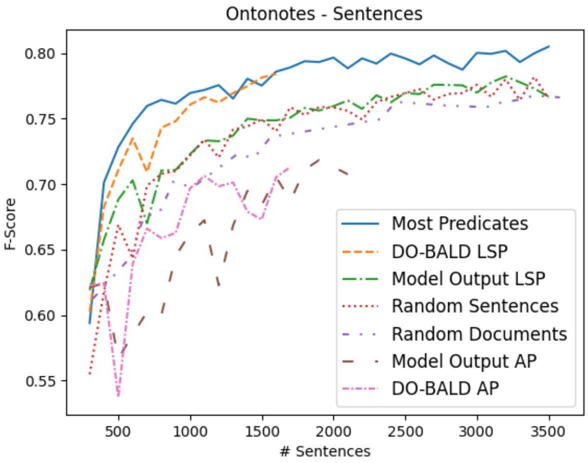

Figure 1: Learning curve of F-score by number of sentences in Ontonotes training data.

7.1 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 for a viable SRL model that can produce sufficiently accurate results to feed into downstream NLP applications.

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 4: 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.

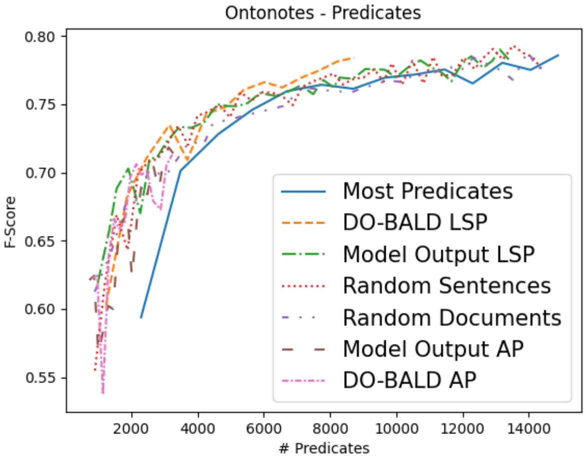

Figure 2: Learning curve of F-score by number of predicates in Ontonotes training data.

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.

On the other hand, the reduction in predicate annotation offered by active learning was more modest. 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.

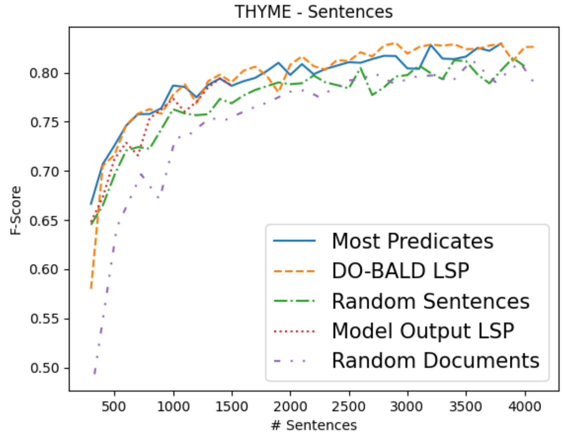

Figure 3: Learning curve of F-score by number of sentences in THYME training data.

The two selection methods that averaged the predicates performed worse than the baselines by sentences. One reason for this may be that the presence of frequent, but easily learned, predicates such as copulas inflating the average confidence of the sentence.

In terms of assessing the impact of whole-document selection, which is necessary for other NLP tasks such as coreference, compared to sampling 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. Sampling individual sentences reduces sentence annotation by 27% and predicate annotation by 20% to reach our benchmark.

7.2 THYME

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

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

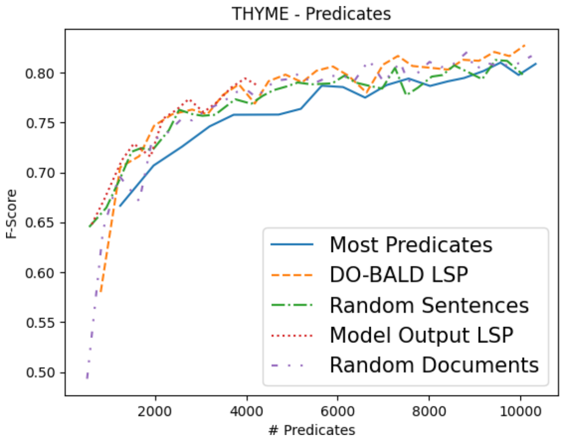

Figure 4: Learning curve of F-score by number of predicates in THYME training data.

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 (4355 predicates) significantly improved by DO-BALD LSP (20% - 4355 predicates) and Output LSP (16% - 3666 predicates), but MostPred is a detriment (30% *more* annotation - 5651 predicates).

8 Conclusions

Between the two proposed methods of aggregating predicate-argument structure scores into a single value to represent a sentence, 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. 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 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. But 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 will have a more substantial impact than reducing the number of predicates.

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.

9 Future Work

Smaller batch sizes per iteration allow more efficient selection of data since the model is updated more frequently and we can reduce redundant information content within the batch that would waste annotation time. Using very small batches is not tractable in tasks that require long model training times. Koshorek et al. (2019) tested selection strategies on randomly sampled batches of data, rather than determining priority of individual instances, but that waters down the benefits of using the selection heuristic. In the future, we plan to investigate ways to balance syntactico-semantic redundancy with the model-based selection techniques in order to improve the learning rate for SRL, while reducing training time for each iteration.

We chose to use a random 200 sentences as our seed set, but the ideal amount and method of selection for active learning for SRL remains an open question. If too few sentences are chosen, or they're not sufficiently diverse, we may encounter the missed class effect (Tomanek et al., 2009), where the model becomes overconfident about instances that greatly differ from what's present in its current training pool, and fails to select them for annotation. On the other hand, selecting too large of a seed set negates the benefits of active learning. In future work we plan to explore unsupervised methods of selecting a semantically diverse seed set. Prior work (Dligach and Palmer, 2011) (Peterson et al., 2014) shows that language models may offer an unsupervised way of selecting rare verb instances and thus beneficial SRL instances.

Acknowledgments

We gratefully acknowledge the support of DARPA AIDA FA8750-18-2-0016 (RAMFIS), NIH: 5R01LM010090-09 THYME, Temporal Relation Discovery for Clinical Text, and NSF ACI 1443085: DIBBS Porting Practical NLP and ML Semantics from Biomedicine to the Earth, Ice and Life Sciences. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of any government agency. Finally, we thank the anonymous IWCS reviewers for their insightful comments and suggestions.

References

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. 2013. Towards comprehensive syntactic and semantic annotations of the clinical narrative. *Journal of the American Medical Informatics Association*, 20(5):922–930.

Jinho Choi, Claire Bonial, and Martha Palmer. 2010. Multilingual Propbank annotation tools: Cornerstone and jubilee. In *Proceedings of the NAACL HLT 2010 Demonstration Session*, pages 13–16, Los Angeles, California. Association for Computational Linguistics.

Dmitriy Dligach and Martha Palmer. 2011. 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. Association for Computational Linguistics.

Li Dong, Chris Quirk, and Mirella Lapata. 2018. 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. Association for Computational Linguistics.

R. Duerr, A. Thessen, C. J. Jenkins, M. Palmer, S. Myers, and S. Ramdeen. 2016. 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.

Yarin Gal and Zoubin Ghahramani. 2016. 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.

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

Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. 2017. 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. Association for Computational Linguistics.

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

Fariz Ikhwantri, Samuel Louvan, Kemal Kurniawan, Bagas Abisena, Valdi Rachman, Alfan Farizki Wicaksono, and Rahmad Mahendra. 2018. Multitask 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. Association for Computational Linguistics.

Omri Koshorek, Gabriel Stanovsky, Yichu Zhou, Vivek Srikumar, and Jonathan Berant. 2019. 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. Association for Computational Linguistics.

Shoushan Li, Yunxia Xue, Zhongqing Wang, and Guodong Zhou. 2013. 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.

Sean MacAvaney, Arman Cohan, and Nazli Goharian. 2017. 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. Association for Computational Linguistics.

Tim O'Gorman, Sameer Pradhan, Martha Palmer, Julia Bonn, Katie Conger, and James Gung. 2018. 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. European Language Resources Association (ELRA).

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

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

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

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

Aditya Siddhant and Zachary C. Lipton. 2018. 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. Association for Computational Linguistics.

Katrin Tomanek, Florian Laws, Udo Hahn, and Hinrich Schütze. 2009. 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. Association for Computational Linguistics.

Chenguang Wang, Laura Chiticariu, and Yunyao Li. 2017. 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.

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. 2013. OntoNotes Release 5.0.

Wen-tau Yih, Matthew Richardson, Chris Meek, Ming-Wei Chang, and Jina Suh. 2016. 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. Association for Computational Linguistics.

Jingbo Zhu and Eduard Hovy. 2007. 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. Association for Computational Linguistics.