# Hypotheses Ranking for Robust Domain Classification And Tracking in Dialogue Systems

*Jean-Philippe Robichaud, Paul A. Crook, Puyang Xu, Omar Zia Khan, Ruhi Sarikaya*

Microsoft Corporation, Redmond WA 98052, USA

{jerobich, pacrook, puyangxu, omarzia.khan, ruhi.sarikaya}@microsoft.com

10.21437/Interspeech.2014-41

## Abstract

We present a novel application of hypothesis ranking (HR) for the task of domain detection in a multi-domain, multi-turn dialog system. Alternate, domain dependent, semantic frames from a spoken language understanding (SLU) analysis are ranked using a gradient boosted decision trees (GBDT) ranker to determine the most likely domain. The ranker, trained using Lambda Rank, makes use of a range of signals derived from the SLU and previous turn context to improve domain detection. On a multi-turn corpus we show that this approach offers accuracy improvements of 3.2% absolute (25.6% relative) compared to relying solely on upfront non-contextual SLU domain models and 2.9% (24.5% relative) improvement even with contextual SLU domain models. We also show that HR can be trained to be robust to changes in the SLU.

**Index Terms**: dialogue systems, natural language understanding, hypothesis ranking, re-ranking, contextual domain classification, lambda rank, gradient boosted decision trees

## 1. Introduction

Natural language interaction, both spoken and typed, is becoming an important user interface approach in a wide variety of scenarios. Many of these scenarios require that spoken language understanding (SLU) can deal with a wide range of tasks and topics. As such, a commonly used architecture first classifies the user's utterance into one of the supported domains (or as an unsupported domain), this is followed by domain dependent intent and slot analysis (*i.e.* intent classification and entity extraction). In such an architecture, although domain classification is potentially less complex than other semantic analysis, any errors made by the domain classifier are significantly more noticeable as they tend to lead to incorrect system actions or responses.

This paper presents the use of ranking, post SLU analysis, which improves domain classification accuracy. The approach tested allows for the easy addition of both linguistic and non-linguistic signals, *e.g.* context, which can further improve overall accuracy. In this formulation, SLU domain, intent and slot analysis is run for all domains. The output of each domain is treated as providing one SLU hypothesis and the set of hypotheses are ranked together.

### 1.1. Related Literature

Approaches based on re-ranking a set of candidate hypotheses have been considered in various aspects of natural language processing. Roark et al., [1] apply max-entropy rankers for sentence boundary detection on $n$-best lists. Shen et al., [2] re-rank $n$-best lists for machine translation of sentences by two techniques, using a variant of the perceptron algorithm and then pairwise ranking by applying classification algorithms. Collins and Koo [3] re-rank the output of a probabilistic parser using a boosting algorithm and providing it with additional features of the parse tree to determine the best sentence parse. Chen et al. [4] improve the accuracy of slot tagging of n-grams by generating multiple possible tags for each n-gram and then using a technique similar to RankBoost. Nguyen et al., [5] use re-ranking algorithms for named entity recognition by using the $n$-best from a CRF and then pairwise classification for ranking.

Various approaches have also been presented for re-ranking for SLU. Morbini et al., [6] present a technique for re-scoring hypotheses by considering the $n$-best from three different speech recognition engines. Each hypothesis is then classified using a maximum entropy classifier for its category (intent within a domain). The 1-best from the intent classifier for each member of the $n$-best is then re-scored using perceptron algorithm with unigram, bigram and trigram features to choose the best hypothesis. Basili et al., [7] also use SVM-based re-ranking of the $n$-best output from the ASR for the more specific purpose of spoken command understanding. In our work, the 1-best output from a single speech recognition engine is processed through SLU models to generate $n$-best list with domain, intent and slots which is then re-ranked. Ng and Lua [8] use a transferable belief model to rank the $n$-best dialog inputs and choose the highest ranked input. Dinarelli et al., [9] use a similar high-level idea of re-ranking the $n$-best models from SLU. In their work, the $n$-best list from the SLU models is pair-wise ranked using SVM kernels which is then followed by a meta-classifier that is used to determine the best output by considering the best results from baseline and re-ranked models. In our work, the $n$-best list is passed through a GBDT with additional features that also represent information across hypotheses as well as previous turns to include contextual information. None of the previous works, cited in this section, exploit contextual features spanning multiple turns to re-rank hypotheses and improve accuracy for domain classification. The approach presented in this paper demonstrates the benefits of using additional features from this type of information.

### 1.2. Lambda Rank Gradient Boosted Decision Trees

Gradient boosted decision trees (GBDT) were introduced by Friedman [10] as a part of a "general gradient descent 'boosting' paradigm ... developed for additive expansions based on any fitting criterion", roughly speaking refining regression tree models through the addition of new trees that move the output error in the direction of steepest gradient decent.

Burges et al., [11] proposed Lambda Rank as an approach to ranking for information retrieval that allows optimisation with non-smooth cost functions; where "the derivatives of the cost with respect to the model parameters are either zero, or are un-

defined." Originally applied to neural network models, Lambda Rank was later applied to GBDT with great success as LambdaMART [12] in the Yahoo! Learning to Rank Challenge.

In this paper, LambdaMART is applied to sets of hypotheses (semantic frames plus knowledge results in this application) to generate a score for each hypothesis which indicates its relative merit (or ranking). LambdaMART learns a model which attempts to reproduce the (unobserved) non-linear function that maps a vector of features that are extracted from each hypothesis to the hypothesis's score such that the model ranks the best hypothesis the highest.

## 2. Re-ranking SLU Semantic Frames

The experimental system architecture is shown in Figure 1. Spoken utterances are interpreted by an automatic speech recogniser (ASR) who's normalised output is interpreted by the statistical spoken language understanding (SLU). Typed input is directly fed to the SLU, bypassing the ASR. For training and testing in this paper, the corpus of utterances are represented as text strings and are fed directly to the SLU component. Within this paper only 1-best ASR input to SLU is considered.

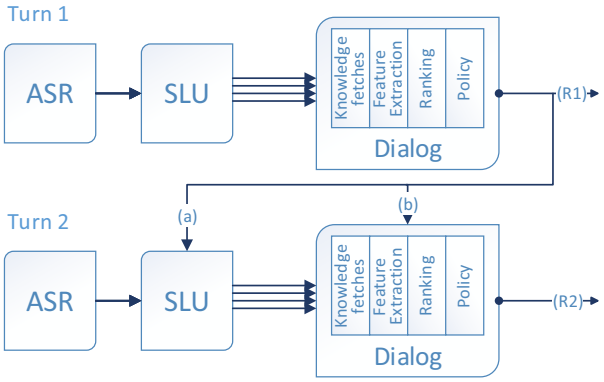


Figure 1: *Schematic diagram of the experimental spoken dialog system where (a) is a domain only contextual signal, (b) is the domain, intent and entities contextual signals, (R1) is selected result of turn 1, (R2) is selected result of turn 2.*

The SLU model is a multi-domain statistical model which follows the domain-intent-slot design outlined in the introduction. It is modular at the domain level, allowing for easy addition of new domains. Support vector machine (SVM) models [13], one per domain, make a binary classification of the utterance with respect to their respective domains. A *domain score* per domain classification is generated by applying a sigmoid function to the output of the SVM, where the output is the distance of the classified example from the SVM decision boundary (with positive examples having a positive output and negative examples a negative output). The output's range is normalised using an affine transform before input to the sigmoid function. After domain classification, intents are then determined using a multi-class SVM intent model, one per domain. Finally entities (slots) are tagged using conditional random fields (CRFs) sequence taggers [14]. The SLU models' input features include n-grams (unigram, bigram, trigram), gazetteers/list of entities and regular expressions.

As shown by Xu and Sarikaya [15], having access to the previous turn's selected domain improves SLU domain prediction accuracy in multi-turn, multi-domain applications. In the SLU used in this work this is implemented using a secondary set of contextual domain models (one per domain) which in addition to the above features have access to the previous turn's domain as selected by the system. When enabled, the output *domain score* of the contextual and non-contextual domain models are mixed through a weighted sum, where the weights are determined using a validation set. The output of an SLU, with such contextual information, can be considered a competitive baseline for our approach.

One variation with the standard domain-intent-slot approach is that in these experiments all of the models for all of the domains are run in parallel as opposed to gating the running of the intent and slot models based on the domain prediction. This has an obvious computational overhead which has to be traded off against the additional information provided. The exploration of this trade-off lies outside the scope of this paper. We only demonstrate the gain in accuracy by having the alternate hypotheses available for re-ranking.

The output of the SLU is a set of semantic frames (SFs), one per domain, which contain intent and slot information, and associated scores. For each semantic frame relevant knowledge, *e.g.* database hits, is fetched and appended. We refer to these assemblies of SFs and knowledge results as *dialogue hypotheses*. Features are then extracted from these dialogue hypotheses which are used as input to the *hypotheses ranker (HR)*. The policy then acts with respect to the ranked list of hypotheses.

The implementation of LambdaMART used in these experiments is very efficient at selecting informative features and we exploit this by extracting close to 1,000 features for HR training and allowing LambdaMART to determine which features to include in the resulting models. In these experiments, the HR models typically learn to use on the order of 100 features. It is worth noting that of the approximately 1,000 extracted features, none contain any word-based features (such as those used by the SLU, *e.g.* no $n$-grams features are available to HR). HR input features include features that span the complete hypotheses list, *e.g.* does a specific slot tagging type occur anywhere in any of the hypotheses, as well as hypothesis specific features such as domain and intent scores, many indicator features, *e.g.* the presence of domains, intents and slot-types, presence of canonicalised entities, and coverage of tagged slots (as percentage of utterance). Contextual HR features include whether the hypothesis's domain matches the top ranked domain from the previous turn, how many slots are in common with the previous top ranked hypothesis, as well as the complete list of previous turn's domains' scores. Some of these features are not available in the first pass through the SLU, on which basis we expect post-SLU ranking to provide benefit. Many of these features will be unique to a particular utterance and/or domains, hence the feature vector of any individual hypothesis is extremely sparse.

Within the context of this paper, features are not generated from the knowledge results, which is a subject of our future work. We consider only the improvement in domain accuracy that can be gained by applying ranking to the SFs' features, *i.e. testing the hypothesis that accuracy improvements can be obtained by ranking dialogue hypotheses based solely on current and previous turn SLU SFs*. The accuracy is measured by comparing the domain of the top ranked hypothesis output by HR and the top scoring domain as output by the SLU against human annotation of the utterance's domain.

### 2.1. Data Sets

The internal data sets used for training, validation and testing mostly comprise utterances collected from real users and crowd workers. It was human annotated for domain, intent and slots.

When drawing training data from the above sources, we ensured there was no overlap between the SLU and HR training sets (to avoid HR becoming overly reliant on the SLU's judgements). The SLU was trained on roughly 70% of the available data. The HR training examples were run through the SLU and feature extraction stages resulting in a set of training examples with input features required by the HR model and with human annotated domain labels as the supervisory signal. For HR a GBDT model was trained using Lambda Rank, as described in Section 1.2. Training and testing of HR was carried out on two versions of the SLU; with and without the optional contextual domain classification models.

In adding the contextual signals to the training and test data there exist two possible sources for populating the previous turn features. These are *(a)* using the domain, intent and slot from the system's previous turn's output, *i.e.* the system's prediction, or *(b)* using the (assumed) more accurate domain, intent and slot for the previous turn as tagged by the human annotators. Contextual data was duplicated to include both versions.

The data sets used for training and testing SLU and HR span 8 distinct domains with multiple intents per domain. The data is split into two distinct types; *(a)* first-turn-only – where no follow up turn was collected, *(b)* multi-turn-possible – where subsequent turns could be collected. The HR training set consist of 52,000 first-turn-only sessions and 6,000 multi-turn-possible sessions. In the latter 49% of sessions have two or more turns, 22% have three or more turns, 11% four or more turns. The test set used has a higher proportion of multi-turn-possible sessions consisting of 9,600 first-turn-only sessions and 4,200 multi-turn-possible sessions. The latter having 50% of session which have two or more turns, 23% three or more turns, 13% four or more turns. Across all the multi-turn-possible data, both training and test sets, it is observed that a domain transition occurs around 27% of the time, *i.e.*, the user switches domain between any two turns of a particular multi-turn session.

### 2.2. Results

Both SLU and SLU+HR were tested using the same held-out test sets and percentage error rates computed by comparing the selected domain with the human annotated one. Results across all test sets are presented in Table 1. The two rows correspond to the two modes of operation of the SLU; with and without the additional contextual domain classification models. Within each row the results for HR correspond to a model that was trained against the matching SLU configuration.

| SLU context | Top SLU | Top HR | Gain | Loss | Net Gain |
|---|---|---|---|---|---|
| none | 9.93 | **8.41** | 2.92 | 1.40 | 1.52 |
| domain | 9.45 | **7.20** | 2.88 | 0.63 | 2.25 |

Table 1: Percentage error rate on all test data. SLU context indicates whether contextual SLU domain models were enabled. HR results based on training with matching SLU model.

HR, based solely on the information available in the current and previous turn SFs, is providing a net gain of 1.52% in domain accuracy when the SLU is operating without contextual domain models and a net gain of 2.25% when SLU is also sensitive to domain context (previous turn domain). Breaking down the net gain into the percentage of utterances that were improved (Gain) and percentage of utterances where HR hurt the domain classification (Loss) we can see that this slightly surprising result is due to the reduction in HR errors when operating with the contextual SLU. As expected the addition of contextual SLU domain models reduce the SLU's overall domain error rate and also reduce the number of examples which HR can correct. (The relatively small improvement seen in the SLU between the two conditions can be attributed to conservative tuning of the contextual SLU, *e.g.* minimisation of the number of errors introduced. Also, by design, SLU contextual domain models have no impact on the first turn.)

### 2.3. Contextual Domain Tracking

The results presented in Table 1 are based on both first-turn-only and multi-turn-possible test sets. Tables 2 and 3 present results obtained exclusively on the multi-turn-possible test set, *i.e.* where contextual influence is more likely to occur.

| SLU context | Top SLU | Top HR | Gain | Loss | Net Gain |
|---|---|---|---|---|---|
| none | 12.58 | **9.37** | 4.29 | 1.08 | 3.22 |
| domain | 11.85 | **8.95** | 3.73 | 0.84 | 2.90 |

Table 2: Percentage error rate considering only multi-turn-possible test set.

| Turn | SLU context | Top SLU | Top HR | Gain | Loss | Net Gain |
|---|---|---|---|---|---|---|
| 1 | none | 7.61 | **7.17** | 0.84 | 0.40 | 0.44 |
| | domain | 7.61 | **6.95** | 0.96 | 0.30 | 0.66 |
| 2+ | none | 16.67 | **11.17** | 7.13 | 1.63 | 5.50 |
| | domain | 15.33 | **10.59** | 6.00 | 1.27 | 4.73 |

Table 3: Percentage error rate considering only multi-turn-possible test set. Reporting first turn and subsequent turns separately.

Focusing on the multi-turn-possible test set only, both the SLU and HR models exhibit increased number of errors compared to the results for all test sets, presented in Table 1. This gives some indication of the increased difficulty in the task of tracking the domain in multi-turn interactions where the domain will not necessarily be explicitly expressed in any one turn and may change between turns. Nevertheless HR shows greater net gain on these examples with 3.22% improvement in the non-contextual SLU condition and 2.9% gain in the contextual domain SLU condition.

Table 3 breaks down these results further presenting percentage error rates separately for the first turn and all subsequent turns in the multi-turn-possible data set. From this table it can be seen that as expected the largest errors occur for the subsequent turns (turn two or later in the dialogue); analysis presented by Xu and Sarikaya [15] indicate that in general domain detection becomes harder for SLU in subsequent dialog turns. As expected, the SLU with contextual domain models improve over the non-contextual SLU but HR provides the most significant net gains, *e.g.* 5.50% with non-contextual SLU and 4.73% net gain with contextual SLU. This is in contrast to the first turn the gains provided by HR which are somewhat limited, 0.44% and 0.66% with the two versions of the SLU, indicating that the SLU is well tuned to the overall task.

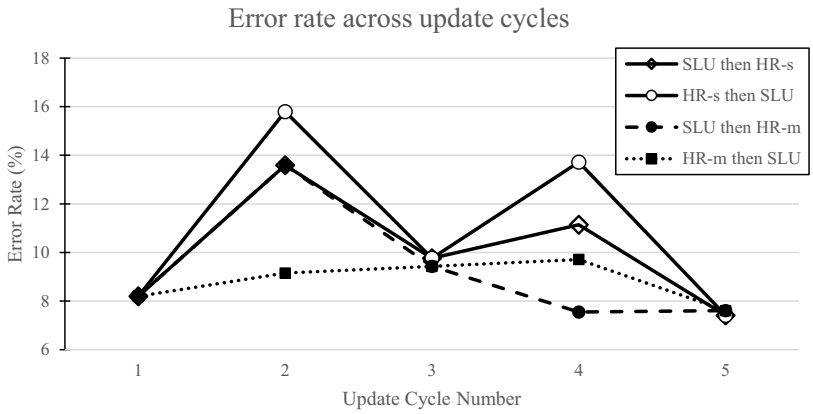


Figure 2: *Evolution of the test set error rates over update cycles where only one model is update at a time. Here, HR-s stands for "HR model built against a single version of SLU models" and HR-m stands for "HR model built against all available previous SLU models"*

## 3. Robustness to SLU Model Updates

A valid concern in architectures comprising multiple statistical models feeding into another set of statistical models is the problem of model update. Technically, if models in upstream stages are updated (in the pipeline discussed in this paper, this refers to the SLU models), the downstream models (the HR model) also requires retraining. However, in industrial environments, there is a strong desire to stage updates to minimise the number of changes performed at once.

To account for this concern, special care was taken in crafting features that are stable across SLU model updates as well as devising a good training strategy for the downstream models. For example, a feature like the rank in the SLU $n$-best of a specific SF is more consistent across updates than the actual raw score of the domain classification SVM models. Similarly, the ratio of the SF domain score to the maximum SF domain score across the entire $n$-best is another example of a stable feature across SLU model updates. Related to this, if the HR model is trained using samples that have been passed through more than one SLU version, the dependence on SLU version specific features decreases.

The GBDT training process provides us with a measure of the information gain provided by each feature. We observe that as the accuracy of the SLU models improves, the relative importance of the features used by the GBDT changes. For instance, the SLU $n$-best rank gains a lot of weight as the SLU accuracy increases. In a similar fashion, the feature that indicates if a SF belongs to the same domain as the previous turn selected domain becomes more important in the HR models as the SLU accuracy improves.

To illustrate the impact of this behaviour, we simulated different deployment strategies in a fictional production system where only one set of models would be allowed to change at one time (see Figure 2). For the "SLU then HR-x" strategies, the updated SLU model is deployed before the retrained HR model gets deployed. It is the other way around for "HR-x then SLU" strategies. As the figure demonstrates, a lower error rate with less variance across cycles can be achieved once the HR model gets trained with multiple prior SLU versions (HR-m) irrespective of the deployment order.

## 4. Discussion and Future Work

Though this paper focused on using SLU signals (both current and previous turn) to enable ranking of the hypotheses, the scope of signals that HR can use is much wider. Positioned toward the end of the system pipeline, HR can make use of signals from further upstream, such as ASR $n$-best and confidence scores, and also knowledge result signals downstream of the SLU, *e.g.* database hits. The latter allows for top-down signals (for example about the existence of an entity) as well as bottom up signals derived from the speech signal to influence the final ranking. This potential expansion in the source of input signals also motivates our investigation into robustness with asynchronous model updates.

In the evaluations presented in this work, the metric used was the accuracy of the predicted domain in the top hypotheses. Further extensions would be to consider the general goodness of the ranking and also dialogue policies that act with respect to the full ranked list. As mentioned in Section 1, there is a trade off in running all the models to re-rank the alternate hypotheses and the accuracy gains received that requires further analysis. In general, nothing in the application of ranking specifically requires the use of GBDT. We also plan to explore using other approaches, *e.g.* RNN, in place of GBDT.

## 5. Conclusions

We presented a novel application of hypothesis ranking for the task of domain detection in a multi-domain, multi-turn dialog system. We show that accuracy improvements can be obtained by ranking dialogue hypotheses based solely on SLU SF current and previous turn information. On a multi-turn corpus we show accuracy improvements of 3.2% absolute (25.6% relative) compared to relying solely on upfront non-contextual SLU domain models and 2.9% (24.5% relative) improvement even with contextual SLU domain models. The majority of this gain being due to more accurate tracking of the domain in second and subsequent turns in multi-turn sessions, *e.g.* 4.75% net gain with the contextual SLU domain models. We also show that HR can be trained to be robust to changes in the SLU models allowing asynchronous model updating.

# 6. References


[1] B. Roark, Y. Liu, M. Harper, R. Stewart, M. Lease, M. Snover, I. Shafran, B. Dorr, J. Hale, A. Krasnyanskaya, and L. Yung, "Reranking for sentence boundary detection in conversational speech," in *Proceedings of ICASSP*, Toulouse, France, May 2006, pp. 545–548.

[2] L. Shen, A. Sarkar, and F. J. Och, "Discriminative reranking for machine translation," in *Proceedings of the Human Lanugage Technology Conference/NAACL*, Boston, MA, May 2004.

[3] M. Collins and T. Koo, "Discriminative reranking for natural language parsing," *Computational Linguistics*, vol. 31, no. 1, pp. 25–70, March 2005.

[4] J. Chen, S. Bangalore, M. Collins, and O. Rambow, "Reranking an n-gram supertagger," in *Proceedings of the Sixth International Workshop on Tree Adjoining Grammar and Related Frameworks (TAG+6)*, Venice, Italy, May 2002, pp. 101–110.

[5] T.-V. T. Nguyen, A. Moschitti, and G. Riccardi, "Kernel-based reranking for named-entity recognition," in *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*, Beijing, China, 2010.

[6] F. Morbini, K. Audhkhasi, R. Artstein, M. V. Segbroeck, K. Sagae, P. S. Georgiou, D. R. Traum, and S. S. Narayanan, "A reranking approach for recognition and classification of speech input in conversational dialogue systems." in *IEEE Workshop on Spoken Language Technology (SLT)*, December 2012, pp. 49–54.

[7] R. Basili, E. Bastianelli, G. Castellucci, D. Nardi, and V. Perera, "Kernel-based discriminative re-ranking for spoken command understanding in hri," in *AI*IA 2013: Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science, M. Baldoni, C. Baroglio, G. Boella, and R. Micalizio, Eds. Springer International Publishing, 2013, vol. 8249, pp. 169–180.

[8] H.-I. Ng and K.-T. Lua, "Dialog input ranking in a multi-domain environment using transferable belief model," in *4th SIGDIAL Workshop on Discourse and Dialogue*, 2003.

[9] M. Dinarelli, A. Moschitti, and G. Riccardi, "Discriminative reranking for spoken language understanding," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 2, pp. 526–539, February 2012.

[10] J. H. Friedman, "Greedy function approximation: A gradient boosting machine." *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 10 2001. [Online]. Available: http://dx.doi.org/10.1214/aos/1013203451

[11] C. J. C. Burges, R. Ragno, and Q. V. Le, "Learning to rank with nonsmooth cost functions," in *Proccedings of NIPS*, 2006.

[12] C. J. C. Burges, K. M. Svore, P. N. Bennett, A. Pastusiak, and Q. Wu, "Learning to Rank Using an Ensemble of Lambda-Gradient Models," *Journal of Machine Learning Research*, vol. 14, pp. 25–35, 2011.

[13] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[14] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of ICML*, 2001.

[15] P. Xu and R. Sarikaya, "Contextual domain classification in spoken language understanding systems using recurrent neural network," in *Proceeding of ICASSP*, 2014.