Hypotheses Ranking and State Tracking for a Multi-Domain Dialog System using Multiple ASR Alternates

Omar Zia Khan, Jean-Philippe Robichaud, Paul Crook and Ruhi Sarikaya

Microsoft Corporation, Redmond WA 98052, USA
{omarzia.khan, jerobich, pacrook, ruhi.sarikaya}@microsoft.com

Abstract

In this paper, we present an approach to improve the accuracy of multi-domain multi-turn spoken dialog system (SDS) by including alternate results from automatic speech recognition (ASR). Often, even if the top ranked result from the ASR is not correct, the correct result may still be available in the NBest list or in the word confusion network (WCN). Thus, the SDS performance can be improved by considering beyond the top ranked choice from the ASR. We employ late binding, such that multiple ASR choices are propagated through the SDS and knowledge fetch so that additional context can be utilized at later stages to determine the top choice that is good for the overall SDS. We rank alternate domain dependent semantic frames, multiple semantic frames per ASR choice, to determine the true SDS output. Using real-world data, extracted from the logs of Cortana personal digital assistant deployed to millions of users, we show that significant gains can be achieved in domain detection, intent determination, and slot tagging, by considering additional results from ASR.

Index Terms: dialog systems, natural language understanding, speech recognition, hypotheses ranking, dialog state tracking, multi-domain classification, contextual domain classification

1. Introduction

Personal digital assistants are gaining more popularity with Siri, Google Now and Cortana being available on different mobile platforms. These assistants typically involve the use of speech for natural language interaction to accomplish various tasks such as controlling the various device functions (e.g., alarm, calendar, note, communication, device management) as well as gaining information related to various entities (e.g., places, weather, or web search). Thus, SDS for such scenarios require multi-domain multi-turn dialog capabilities. Recent advances in speech recognition have significantly improved the accuracy of ASR, however, it is still possible that the top ASR choice may not be correct which can adversely impact spoken language understanding (SLU) and dialog response. The use of multiple alternates from the ASR (NBest list or word confusion network) in addition to the top alternate can improve the performance of SDS as the correct output is often available in this list of alternates even if the top choice is incorrect. Access to these ASR alternates downstream in SDS stack can help because additional context and knowledge can be exploited downstream to rerank multiple ASR alternates and improve the overall accuracy.

In this paper, we present an approach that considers alternate choices provided by the ASR to improve domain detection, intent determination and slot tagging. More specifically, we adopt the paradigm proposed in [1] that generates an alternate hypothesis per domain during the SLU analysis, and then these hypotheses are reranked post-SLU analysis by considering additional context and knowledge features to determine the correct domain. We extend this approach to inject additional hypotheses corresponding to each ASR alternate, which are then reranked post-SLU analysis to determine the correct choice. Using multiple alternates per domain also allows us to determine the correct semantic frame (combination of domain, intent and slot) rather than only predict the correct domain as in [1].

2. Related Literature

The idea of using multiple ASR alternates in SLU and SDS has been widely considered to reduce the effect of mistakes resulting from only considering the top ASR choice. These alternates may be in the form of an NBest list produced by the ASR in addition to the top choice. They can also be in the form of a word confusion network (WCN) that can then be considered either on its own or otherwise to generate alternates as required. Stolcke et al., [2] show that the word error rate for ASR can be reduced by using NBest lists. In [3] it was shown the WER can be further reduced by considering the use of a recognition lattice. Erdogan et al., [4] use semantic analysis to improve the accuracy of the ASR by using lexical and semantic information and show reduction in WER. Lopez-Cozar and Callejas [5] use a technique to correct the output of an ASR by applying various semantic, syntactic and lexical patterns,most of which are provided by domain experts, on multiple ASR alternates. Hazen et al., [6] use multiple alternates, along with their confidence scores, from ASR to extract features that help improve SLU. Various proposals have been presented to improve SLU by considering WCN rather than NBest lists since they can provide additional alternates and have a higher oracle accuracy [7, 8, 9, 10, 11]. Multiple ASR results have also been used for dialog state tracking [12, 13] as well as explicitly tracking hypotheses based on NBest list [14], reranking NBest hypotheses in dialog based on linear regression [15], and reranking NBest hypotheses based on contextual features [16].

Most of the above approaches employ joint optimization by considering the speech and knowledge results together. As argued by Calvo et al., [17] this can result in an excessively large search space and requires proper weighting for each component. Therefore, we adopt a modular approach where multiple ASR results are fed into the SLU and are then individually ranked.

The above work cited on using multiple ASR results in SLU and dialog state tracking performs SLU for a single task or domain. Planells et al., [18] and Robichaud et al. [1], have described the challenges in constructing a multi-domain dialog system that integrates heterogeneous spoken dialog systems. In this paper, we demonstrate that the use of multiple ASR results is also helpful for such multi-domain multi-turn SLU and dialog state tracking, where it is not known a priori which domain/task the user may want to complete. This is also the first study that applies this approach to a truly web-scale application that handles millions of queries everyday.

3. Using Multiple ASR Results in SLU

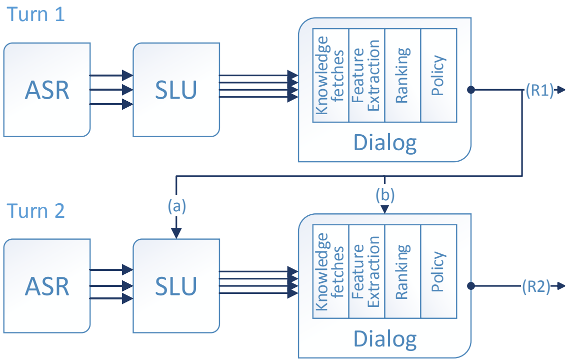

Figure 1: *Architecture of the SDS where (R1) is selected result of turn 1, (R2) is selected result of turn 2, and contextual signals related to domain, intent and entities from turn 1 are passed to (a) SLU and (b) Dialog engine*

The high-level system architecture is shown in Figure 1. NBest results from ASR are fed in to the SLU analysis module (where N is the number of ASR alternates). The SLU analysis module generates NxM semantic frames (where M is the number of domains) that are fed in to the dialog engine. The dialog engine executes knowledge fetches and performs feature extraction to rank the NxM hypotheses and output the top choice. For multi-turn queries, the results of the previous turn are also made available to the system as it has been shown to improve the accuracy in multi-turn, multi-domain scenarios [19, 20].

The SLU analysis module includes several components. The first component is responsible for domain classification using a binary support vector machine (SVM) classifier per domain. This allows new domains to be added or existing domains to be retrained in isolation without impact on other unrelated domains. For each domain, we also classify the intent using a multi-class SVM classifier. Additionally, we also have component that performs slot tagging (entity extraction) using conditional random field (CRF) sequence taggers, with possibly multiple slots per utterance. The models use linguistic signals (unigrams, bigrams, trigrams, regular expressions, dictionaries of domain specific entities) as well as contextual signals (domain detected in previous turn, system response to previous turn output). The output of the SLU analysis module is one semantic frame per domain per ASR input. The semantic frame comprises the combination of domain, intent and tagged slot value pairs along with a confidence score for this combination.

The output of the SLU analysis module is propagated through the dialog engine to augment each semantic frame with any available knowledge fetch results as well as extracted features. As in [1], we name these augmented semantic frames as dialog hypotheses and a hypotheses ranker (HR) is responsible for ranking alternate dialog hypotheses.

The key difference of our proposed approach compared to the system described in [1] is that feeding multiple NBest ASR inputs to the SLU analysis module and receiving back a list of NxM semantic frames allow the hypothesis ranker to rank the expanded the hypotheses by optimizing for the accuracy of semantic frame rather than only predicting the correct domain. This is possible because multiple semantic frames may be available per domain due to the use of NBest ASR results (with only the top choice from SR, there is only one semantic frame per domain available).

The hypotheses ranker uses an internal implementation of LambdaMART [21], based on the concept of Gradient Boosted Decision Trees [22], that ranks the various dialog hypotheses. Any other ranking technique that can learn a (possibly non-linear) model to map from a vector of features for each hypothesis to a relative score for that hypothesis, given a list of hypotheses, could be used. The objective of our ranker is to ensure that the semantic frame (i.e., domain, intent and slot) is correct. Thus, the order of ranking is 1) domain, intent and slots are correct, and if no such hypothesis is available then, 2) domain and intent, or domain and slots, are correct, 3) domain is correct, 4) all slots are correct, and finally 5) everything else.

The accuracy of HR on the ASR 1Best is treated as the baseline. We aspire to the accuracy of HR using transcription, with no ASR errors. A more realistic upper bound is the HR accuracy of an oracle that can pick the correct ASR output if it is available in the NBest list and choose ASR 1Best otherwise. We measure domain precision and recall, intent accuracy, slot F1 score as well as the semantic frame accuracy. The semantic frame is considered correct if all domain, intent and slots are correct for a given utterance, and incorrect otherwise.

The SLU analysis for each ASR alternate can be performed in parallel so this design doesn't result in latency overhead. We do not consider any optimizations for any capacity constraints as that is considered external to the scope of this paper.

3.1. Feature Extraction

The implementation of LambdaMART used for experiments is capable of efficiently selecting relevant features. This allows us to extract close to 1500 features for the purpose of training and we let LambdaMART choose features that are useful for ranking dialog hypotheses (around 250). We include all features specified in [1]. These include 1) features specific to the hypothesis such as the domain/intent scores, indicator features to indicate presence/absence of different types of slots, and coverage of tagged slots, 2) features relevant to the hypothesis list (the list comprising one semantic frame per domain) such as whether a particular semantic tagging occurred anywhere in the hypotheses list, and 3) contextual features such as whether the domain from the current hypothesis matches the previous turn output. In addition, we add features that encode information available from NBest list. The following are a list of the types of features from ASR NBest list that we use:

- Features specific to the ASR alternate from which the dialog hypothesis originated. These can include the incoming rank, the confidence score, the acoustic model score and the language model score of the ASR alternate. Additionally, we also consider the SLU analysis scores weighted by the ASR confidence.
- Features that encode information about the NBest list, such as size of the list, confidence (numerical or categorical such as High/Medium/Low) of the top choice, and the relative difference and ratio of the top vs. subsequent ranked ASR choices.
- Features that indicate the agreement or diversity in the

Table 1: Statistics to describe the data set used in experimental evaluation

Data Set	Total Turns	% of 2+ Turns	Mean NBest	1Best Incorrect	Oracle Correct	WER
Train	30,653	12.7%	3.4	22.3%	83.9%	24.2%
Test	8,914	42%	3.7	25.9%	75.3%	26.8%

Table 2: Aggregated Results of HR accuracy using multiple ASR alternates across all domains and all turns

Model	Domain	Intent	Slot F1	Semantic Frame
Transcription	98.1	93.3	85.3	81.5
Oracle	95.0	87.3	72.9	67.1
1Best	93.9	85.0	67.7	61.1
NBest	95.7	86.4	67.9	63.3

SLU analysis of different ASR alternates. Examples of these features include the SLU score of hypotheses from various domains, the presence of the same slot type or slot value across SR alternates in the same domain, or the diversity in the SLU results when comparing them across ASR alternates expressed using Gini impurity [23].

- Features that encode information about the top ranked SLU alternate (winning hypothesis per alternate) from each ASR alternate, as if it was considered independently. Examples of such features include number of winning hypotheses per domain, the diversity of domains in winning hypotheses expressed using Gini impurity, or the agreement among the winning hypotheses on the top ranked semantic frame.
- Features that indicate the agreement or diversity of an SR alternate with the top ASR choice, such as the difference in the domain, intent and slot value pairs.

Note that none of the features used by HR are word-based (e.g., no ngram or regular expression based features are used by HR) as they are assumed to have been already covered in the SLU analysis module. For the results presented in this paper, post-knowledge features (database hits, access to user personalization such as address book, locations and music library, or the possible Cortana response for a given alternate) were not used to train HR. The addition of such features, simple to add in this paradigm, is expected to further improve the performance and is left as future work.

3.2. Data Set

Data is sampled from logs collected from usage of real-world Cortana users. Anonymized audio is transcribed to receive the true user query. These transcriptions are annotated for SLU (domain, intent and slots) in a contextual manner, with the annotators provided access to previous turns. The transcription and ASR alternates for each query are independently propagated through the SDS pipeline to perform feature extraction. We use the labels generated by the annotators as ground truth and train HR such that it can output the correct dialog hypothesis from a list of dialog hypotheses (including dialog hypotheses from multiple ASR results). For evaluation, we use a held-out set of the transcribed and annotated data to test the performance of the ranker. No development set is used as no model parameter tuning was performed for the experiments described here.

The data set used had 9 distinct domains with over 188 intents and 142 distinct slots. Table 1 describes various statistics related to the data set used for experiments. We can see over 30,000 utterances are used for training purposes and around 9,000 utterances are used for testing. The average size of the NBest list available for theses utterances in the logs is 3.4. Both training and test sets have multi-turn queries, though a bit more in the testing set. It is worth pointing out here that the ASR alternates available in the logs are in Display form (e.g., NE 12th) whereas the transcriptions are manually transcribed in a lexical form (e.g., North East Twelfth). We use an internal tool to perform text normalization and subsequently inverse text normalization (TN/ITN) on the transcribed utterances, but do not manually massage them to match the exact Display Form available in the logs, which is used for presenting the ASR output on screen to the user. This explains the higher than expected percentage of utterances for which the top ASR result does not match the transcription and high WER as well as the relatively low oracle match rate in Table 1.

3.3. Results

Table 2 shows results of our approach based on using multiple ASR alternates compared with transcription, the ASR oracle, and the top ASR alternate. We can see that the model improves for domain (1.8%) and intent (1.4%) and semantic frame (2.2%). However, there is less gain in the slot F1 score (0.2). The NBest performance for domain even exceeds that of the oracle accuracy for domain. This is possible because in some cases the exact query matching user input is not in the NBest but another slightly different query from the NBest list is chosen by the model which has the correct domain. We see that 38.3% of the possible oracle improvement is being recovered by HR for semantic frame accuracy. Most of the remaining possible gains are dependent on improvements in slot tagging. We are currently exploring additional features based on knowledge fetch (such as personalizing the ranking to the user by considering presence of names in the user's address book or the songs in personal music library, as well as third-party knowledge results such as whether the value of particular slot in a weather or place domain are really a place name or business name).

3.3.1. Sample Results from HR using ASR Alternates

We present some positive and negative examples of utterances below where the HR with multiple alternates improves on the top ranked ASR choice or otherwise is unable to disambiguate the correct alternate in the NBest list. The correct output is the transcription, the top ASR output is labeled as SR1, and the NBest ranker output is italicized:

- {Transcription: "Drive home", SR1: "I'm home", SR2: "*Drive home*" }. In this case, the higher domain score for SR2 helps in correctly reranking the alternates.
- {Transcription: "Try again", SR1: "Dragon ball", SR2: "Dragon age", SR3: "Dragon", SR4: "*Try again*"}. For multi-turn queries, there is a higher chance that the user is engaging in clarification, repetition, confirmation, rejection or selection that helps HR pick SR4 as the output.

Table 3: Aggregated Results of HR accuracy using multiple ASR alternates across all domains separated by turns

Model	Domain		Intent		Slot F1		Semantic Frame	
	Turn 1	Turns 2+	Turn 1	Turns 2+	Turn 1	Turns 2+	Turn 1	Turns 2+
Transcription	98.1	98.2	96.6	88.8	88.3	81.3	88.0	72.7
Oracle	93.5	97.1	89.9	83.9	77.5	66.9	73.2	58.8
ASR 1Best	92.0	96.9	87.5	81.6	72.6	61.3	67.0	53.2
ASR NBest	94.8	97.1	88.5	81.8	72.8	61.3	70.5	53.4

Table 4: Results of using multiple ASR alternates in ranking HR for Communication domain

Model	Precision	Recall	Intent	Slot F1	SF
Transcription	97.9	96.2	90.6	87.4	88.2
Oracle	94.3	91.1	80.9	68.5	67.3
1Best	93.6	89.1	76.9	62.9	62.7
NBest	95.8	94.3	81.1	63.1	68.1

- {Transcription: “Driving directions to Costco”, SR1: “*Driving directions to Cosco*”, SR2: “Driving directions to Costco”}. In this case, the model sticks with the SR1, even though SR2 is the correct choice. To avoid such issues, features based on knowledge-fetch (database hit for Costco as a place name and ability to formulate driving directions for Costco and not Cosco) are needed. We are extending our approach to use such features.
- {Transcription: “What’s the temperature like in Celsius”, SR1: “What’s the temperature in Celsius”, SR2: “*What’s the temperature in San Jose*”, SR3: “What’s the temperature in San Jews”, }. In this case, the SR1 is correct but the model incorrectly switches to SR2.

Most of the regressions of HR using multiple alternates are related to slots especially with a slot being added, removed or modified. We anticipate that the use of features based on knowledge fetch will also help reduce such mistakes.

3.3.2. Analysis of Turn 1 vs Turn 2+ Results

Table 3 presents the results of using multiple ASR alternates segregating first turn and 2+ turn queries. We can see that the overall trend of improvement in the accuracy of domain prediction, intent determination and semantic frame accuracy persists across first and 2+ turns. We can see a couple of interesting things to note. The domain accuracy is slightly higher for 2+ turns for oracle, 1Best and NBest. This can be explained by the nature of some shorter prompts that are present in 2+ turns that involve confirmation, rejection or selection of items. If the top ASR is incorrect, they are usually present in the NBest list. We do see bigger differences in intent determination, slot F1 and semantic frame accuracy with the first turn results being considerably higher. This can be due to the cascading nature of errors where if the previous turn is incorrect, the errors will propagate across turns impacting intent and slot more than domain.

3.3.3. Analysis of Results for Individual Domains

We also analyze the results across 3 different Cortana domains. These domains are Communication (calling and texting) , Calendar (create, edit, view , delete appointments) and Device Control (apps, music, and settings). The results for these domains are respectively shown in Tables 4, 5, and 6. We see that we get an improvement of 4.4% for communication, 1.2% for calendar and 0.7% for device control on semantic frame accuracy. We also see gain in domain precision and domain recall, intent accuracy and semantic frame accuracy for almost all domains (accuracy is similar on calendar intent).

Table 5: Results of using multiple ASR alternates in ranking HR for Calendar domain domain

Model	Precision	Recall	Intent	Slot F1	SF
Transcription	96.1	97.4	85.8	83.2	73.4
Oracle	89.1	96.6	81.3	73.1	64.9
1Best	87.4	96.6	81.2	68.1	61.2
NBest	94.1	96.3	81.1	71.2	62.4

Table 6: Results of using multiple ASR alternates in ranking HR for Device Control domain domain

Model	Precision	Recall	Intent	Slot F1	SF
Transcription	97.6	99.5	96.5	86.1	90.9
Oracle	91.8	95.0	89.4	53.5	70.1
1Best	89.9	94.0	85.9	38.6	62.7
NBest	91.5	95.7	86.3	38.7	63.4

4. Conclusions and Future Work

In this paper, we have demonstrated that multiple ASR alternates can improve the robustness of multi-domain, multi-turn SDS. We used a data set extracted from the logs of Cortana to show absolute gains of 1.8% on domain detection, 1.4% on intent classification and 2.2% on semantic frame accuracy (38.3% of the possible gain considering an ASR oracle). Most of the gains are due to correctly analyzing the domain and intent. In a multi-domain SDS, it is imperative to reduce domain and intent errors, otherwise it can initiate an incorrect task. Once the correct task from the correct domain has been initiated, it is easier to recover from potential errors in slot tagging in subsequent turns.

To close the gap between the semantic frame accuracy of HR using multiple alternates with that of the oracle, we need to improve slot tagging. We plan to extend our approach by adding features based on knowledge-fetch results that can help slot tagging by resolving slot values in the NBest list. We also plan to investigate increasing the potential for improvement by considering alternates from WCN instead of an NBest list. The alternates from a WCN can be significantly more and knowledge fetch for multiple alternates can be expensive, so we are also examining the possibility of pruning NBest alternates for which knowledge results are fetched to minimize any latency or capacity concerns.

5. References

[1] J.-P. Robichaud, P. Crook, P. Xu, O. Z. Khan, and R. Sarikaya, "Hypotheses ranking for robust domain classification and tracking in dialogue systems," in *Proceedings of INTERSPEECH*, 2014.

[2] A. Stolcke, Y. Konig, and M. Weintraub, "Explicit word error minimization in N-Best list rescoring," in *Proceedings of European Conference on Speech Communication and Technology*, 1997.

[3] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus among words: Lattice-based word error minimization," in *Proceedings of EUROSPEECH*, 1999.

[4] H. Erdogan, R. Sarikaya, S. F. Chen, Y. Gao, and M. Picheny, "Using semantic analysis to improve speech recognition performance," *Computer Speech & Language*, vol. 19, no. 3, pp. 321–343, 2005.

[5] R. Lopez-Cozar and Z. Callejas, "ASR post-correction for spoken dialogue systems based on semantic, syntactic, lexical and contextual information," *Speech Communication*, vol. 50, no. 8–9, pp. 745–766, 2008.

[6] T. J. Hazen, S. Seneff, and J. Polifroni, "Recognition confidence scoring and its use in speech understanding systems," *Computer Speech and Language*, vol. 16, no. 1, pp. 49–67, 2002.

[7] G. Tur, J. Wright, A. Gorin, G. Riccardi, and D. Hakkani-Tr, "Improving spoken language understanding using word confusion networks," in *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, 2002.

[8] D. Hakkani-Tur, F. Bechet, G. Riccardi, and G. Tur, "Beyond ASR 1-best: Using word confusion networks in spoken language understanding," *Computer Speech & Language*, vol. 20, no. 4, pp. 495–514, 2006.

[9] M. Henderson, M. Gasic, B. Thomson, P. Tsiakoulis, K. Yu, and S. Young, "Discriminative spoken language understanding using word confusion networks," in *Proceedings of Spoken Language Technology (SLT) Workshop*, 2012.

[10] A. Deoras, G. Tur, R. Sarikaya, and D. Hakkani-Tur, "Joint discriminative decoding of words and semantic tags for spoken language understanding," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 21, no. 8, pp. 1612–1621, 2013.

[11] J. Łvec, P. Ircing, and L. Łmdl, "Semantic entity detection from multiple ASR hypotheses within the WFST framework," in *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2013.

[12] S. Young, M. Gasic, S. Keizer, F. Mairesse, J. Schatzmann, B. Thomson, and K. Yu, "The Hidden Information State model: A practical framework for POMDP-based spoken dialogue management," *Computer Speech & Language*, vol. 24, no. 2, pp. 150–174, 2010.

[13] J. D. Williams, "Challenges and opportunities for state tracking in statistical spoken dialog systems: Results from two public deployments," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 8, pp. 959–970, 2012.

[14] J. D. Williams, "Exploiting the ASR N-Best by tracking multiple dialog state hypotheses," in *Proceedings of INTERSPEECH*, 2008.

[15] A. Chotimongkol and A. I. Rudnicky, "N-best speech hypotheses reordering using linear regression," in *Proceedings of EUROSPEECH*, 2001.

[16] R. Jonson, "Dialogue context-based re-ranking of ASR hypotheses," in *Spoken Language Technology (SLT) Workshop*, 2006.

[17] M. Calvo, F. Garcia, L.-F. Hurtado, S. Jimenez, and E. Sanchis, "Exploiting multiple ASR outputs for a spoken language understanding task," in *Proceedings of The Seventeenth Conference on Computational Natural Language Learning,*, 2013.

[18] J. Planells, L.-F. Hurtado, E. Segarra, and E. Sanchis, "A multi-domain dialog system to integrate heterogeneous spoken dialog systems," in *Proceedings of INTERSPEECH*, 2013.

[19] A. Bhargava, A. Celikyilmaz, D. Hakkani-Tur, and R. Sarikaya, "Easy contextual intent prediction and slot detection," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.

[20] P. Xu and R. Sarikaya, "Contextual domain classification in spoken language understanding systems using recurrent neural networks," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.

[21] C. J. 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.

[22] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2000.

[23] M. A. Gil and P. Gil, "A procedure to test the suitability of a factor for stratification in estimating diversity," *Applied Mathematics and Computation*, vol. 43, no. 3, pp. 221–229, 1991.