Position paper presented at CONVERSATIONS 2022 – the 6th International Workshop on Chatbot Research, Applications and Design, hosted by the University of Amsterdam, Netherlands, November 22-23.

Assessing the Natural Language Understanding Dilemma of Chatbots: Seeing or not Seeing the Forest for the Trees

Esmé Manandise[0000−0003−2752−4894] and Raj Srivastava[0000−0003−3155−4957]

[1] Intuit AI Research Center
Marine Way. 2632,
Mountain View, Ca 94043, USA
esme_manandise@intuit.com
[2] Georgia Institute of Technology
Atlanta, GA 94043, USA
rsrivastava60@gatech.edu

Abstract. Customer-service chatbots built on intents capture the meaning of utterances by aligning utterances with pre-defined domain labels curated by humans to trigger responses towards some goal within an application. In intent classification, content words are preferred over derivational and inflectional morphemes as well as functional words. With preprocessing tasks like lemmatization and stopword deletion, morphemes and functional words are suppressed from models as unnecessary language details that contribute little or no meaning. In this position paper, we report on a pilot study to better understand chatbot language failures at utterance level. To what extent is the inability of a chatbot to understand customers' queries affected by standard processes like lemmatization or stopword preprocessing? While the utterances *What is a tax refund?* and *What is my tax refund?* are about *refund* (intent forest), the definite determiner *a* and the possessive *my* (each a tree species) point to different question types—definitional and customer-centric, respectively. Each should warrant different answers and policies to supply the answers. Our preliminary findings suggest that intent classification would benefit from models inclusive of derivational and inflectional morphemes as well as functional words.

Keywords: Chatbot · Deep Parsing · Failures · Intents · Lemmatization · Language Features · Natural Language Understanding · Shallow Parsing · Stopwords · TurboTax Assistant.

1 Introduction

Text-based customer-service chatbots are software applications that enable customers to request help and have queries answered in a conversation-like manner instead of interacting with live agents. Typically, they are domain-specific and thus specialized to provide answers relevant to the domain at hand. Ideally, these chatbots simulate how humans converse about the domain topics: how and which information is gathered and conveyed, how questions are to be interpreted and answered. They are designed to streamline conversational customer interactions [1,2,3,5,7]. However, one source of endless customer frustration is the failure by a chatbot to understand a customer's query [4,8,9,10]. For example, after submitting the query *What is my tax refund?* to a tax-domain customer-service chatbot, a customer reacts to the chatbot answer with *dont care about tax refund in general. Google can tell me that! Want to know what is MY refund!!!!!.* While the chatbot recognizes *refund* as intent when answering with a definition of *tax refund*, it fails to understand that the query of *refund* relates to the customer scenario as indicated by the possessive *my*.

With traditional intent classification models, meaning is largely derived from the content words present in the queries. To better understand the conversational failures of a specific chatbot, namely TurboTax Assistant, we embarked on an **exploratory** study to determine the meaning import of non-content words (functional words) and morphemes in query understanding.

2 TurboTax Assistant (TTA) Overview

TurboTax customers engage in writing and in English with TTA to get help on various product-, tax- and customer-specific questions. The current TTA is intent-based. TTA is embedded in the TurboTax Online product as a chat application. TTA receives the customer query as text and uses Dialogflow as the natural-language-understanding (NLU) engine to map the query to predefined domain intents. TTA has access to customers' filing status to answer queries and to surface possible *next intent* suggestions. The Convo Design tool enables content designers to add domain rules into the TTA system based on filing status before and after the NLU engine maps a query to an intent. If the customer requests to contact a live agent, TTA files a support ticket with a written explanation from the customer. The request is then routed by a topic algorithm to the appropriate customer support queue given the explanation. The customer is scheduled for a callback from a support agent.

3 Content Words, Functional Words and Morphemes

Intent-based text-based chatbots rely on content words as the source of meaning for customers' queries. With intents, **meaning is an approximation of what is said** and it is considered sufficient to trigger a response. The language models are built on nouns and verbs with the exception of some adjectives and adverbs.

For our discussion, assume that each of the raw queries for the chatbot are represented as in column *Schematic Inputs* of Table 1. The function words and inflections like determiners, conjunctions, prepositions, auxiliaries, possessives, plurals, tenses, etc. are stripped from the inputs to be processed for *meaning.*

Queries 1 and 2 share identical chatbot representations, which prompts TTA to assign an intent of status update to both queries. However, only query 1

Table 1: Sample of Real-Time Raw and Simplified Queries.

	Raw Queries	Schematic Inputs
1	why didn't I get my refund?	why + not + get + refund
2	why didn't I get a refund?	why + not + get + refund
3	should I list an adult child who is dependent?	list + adult + child + dependent
4	how to add a w-2 to my already filed return	how + add + w-2 + file + return
5	personal information changed	personal + information + change
6	change personal information	change + personal + information
7	I should claim my adult child on return?	claim + adult child + return
8	I claimed my adult child on return	claim + adult child + return

is a request for a status update. To generate appropriate answers, TTA would have to leverage the meaning from the possessive *my* and from the indefinite determiner *a*, respectively. *my* functions as a definite operator that points to a referent *refund* whose existence is assumed as fact by the customer. In (2), the indefinite quantifier *a* in the scope of the negation *not* is an indirect request to explain why the customer did not qualify for a refund, which involves checking the customer's tax scenario.

With query 3, TTA cannot classify the intent and returns links for information about adult children and taxes. The reply from the customer is *I want to know if best for me.* TTA fails to detect that the query is about the best course of action for the customer given the customer tax scenario. *Should* conveys the sense of recommendation in the original raw query.

With query 4, TTA no longer has the tense/time features *-ed* and *already* available that point to a past action and a state of fact. TTA explains how to add W2 as if filing was not done yet.

In query 5, the removal of *-ed* triggers TTA to interpret the utterance as a request on how to change personal information on the TurboTax account instead of consuming the input as a fact that the information has changed. Queries 5 and 6 have similar interpretations of how-to.

Finally, with the stopword removal of the modal *should* in query 7 and the lemmatization of *claimed* to *claim* in query 8, TTA interprets both queries to be about dependent filing. However, query 7 seeks a recommendation on whether the customer should do it given the customer's tax scenario while query 8 states a fact about the customer having claimed an adult dependent when filing.

The examples in Table 1 above suggest that non-content language units and functional words substantively contribute to what customers intend to convey in their queries([9]).

4 Methodology Overview

To scope the contribution of non-content words and morphemes to query understanding, we approach the problem as a binary classification task:

1. **Shallow** means content-word-based intent classification is sufficient for TTA to understand a customer query.

2. **Deep** means deeper feature-based semantic parsing is required to surface meaning for TTA.

Consider some further candidates below (Table 2).

Table 2: Utterance Candidates for Shallow versus Deep Parsing.

	Shallow Parsing	Deep Parsing
1	live help	my refund sounds lower than last year
2	why did you ask questions about my husband	why is my refund so low
3	what is NYPFL	are you sure you calculated my taxes correctly
4	what is a tax refund	what is my tax refund
5	how to start over	how to recalculate my wrong refund

4.1 From Data Observations to Data Labeling

We used two separate domain corpora for statistically-based corpus analyses to determine the salience of features: (1) a collection of tax forms and instructions, and (2) a collection of TTA utterances. This TTA corpus is separate from the TTA corpus used for the shallow-deep classification task. We isolate the following 10 feature classes:

- Negation (*not, never, nontaxable ...*)
- Temporal expressions (duration versus punctual)
- Verb tenses
- Wh-words (*why, where, what ...*)
- Possessives
- Comparisons
- Temporal and/or spatial prepositions
- Quantifiers (*a, any, some, none ...*)
- Number of tokens in query
- Number of multiword expressions in query (*climate leadership adjustment rebate, nonrefundable tax credit rate*)

Most of these features account for the meaning that is lost when standard intent classification uses lemmatization and stopwords, or focuses exclusively on content words. These features hold enough meaning to significantly change the frame of customer queries as described in Section *Content Words, Functional Words and Morphemes*. Given these observations, data is labeled *deep* if the query contains some combination of function words and/or language features. Otherwise, it is labeled *shallow*.

For our corpora, mutual information scores and random forest feature weights indicate that some of the most salient features are:

- *num_poss* - number of possessive pronouns
- *num_future* - number of future tense verbs
- *contains_wh_word*
- *contains_time_term* - indicates presence of a temporal expression (*before, after, last* ...)
- *num_mwe* - number of terms or multiword expressions (determined through dependency parsing, examples include *post office* and *phone number*)
- *num_tokens*

For **feature evaluation**, we rely on the following 6 methods:

- Ridge regression
- Mutual information
- Feature weights of classifiers
- Feature correlations with each other
- Feature correlations with labels
- Observations from data labeling

Note that, while some features performed poorly, these feature evaluation methods give scores dependent on the data distribution. Therefore, features that seem weak like *num_future* compared to others are likely still important for customer queries under certain contexts.

4.2 Shallow-Deep Classification

This binary classification task uses the labels *shallow* and *deep* to indicate whether a customer query needs shallow or deep understanding. In a customer-service scenario, it is much more damaging for a deep query to be misclassified as shallow compared to the other way around, so our metric of choice is that of **recall** to minimize false negatives; thus, **deep** is considered the positive label.

The **three models** used were **random forest** (nonlinear method), **logistic regression** (linear method) and **ridge regression** (primarily for feature evaluation). All three have easily-interpretable feature weights. Cross-validation and hyperparameter search were used as well.

We built 2 classifiers. Classifier 1 was trained on 2,000 of the 2,500 gold-standard samples manually-labeled by 2 developers as shallow or deep using the set of language features. Random forest, logistic regression, and ridge regression were evaluated on the remaining 500 gold-standard samples. Random forest performed best. Thus, Classifier 1 is a random forest, trained and tested completely on different sections of the gold standard. Then, Classifier 1 was used to predict shallow or deep for 50,000 utterances. Classifier 2 is a random forest trained on the 50,000 samples that were labeled by Classifier 1. The purpose of Classifier 2 was to evaluate training data size vs. performance; so, Classifier 2 was trained on 5,000, 10,000, ..., 50,000 training samples and evaluated on the gold standard–all 2,500 samples. Classifier 1 has more validity to its results while Classifier 2 was more of a proof of concept that lacks some validity. As seen in Fig. 1, random forest has the best performance, slightly outperforming logistic regression. Hyperparameter search in random forest also tends to select for a low number of estimators, indicating it may have *overfit* less than logistic regression.

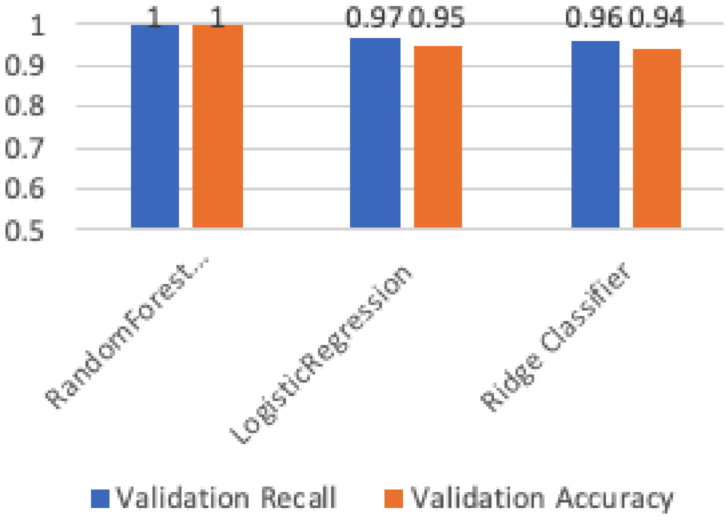

Fig. 1: Performance of Classifiers.

4.3 Insight on Data Distribution Analysis

The data distribution of the training samples can be seen in Table 3. Almost 75% of queries are deep, suggesting the need for deeper query understanding.

Table 3: Data distribution of the 50,000 training samples with Classifier 2.

Label	Number of Samples	Percentage
Shallow	12582	25.2%
Deep	37418	74.8%

5 Concluding Remarks

The language features that determine if a customer's query requires deeper understanding include a variety of dimensions, in isolation or together. The model performs at 98% accuracy and 98% recall on the gold-standard dataset. Thus, the model and data support the idea that shallow intent-based understanding is sufficient to trigger responses for some type of utterances while other utterances need deeper semantic parsing. Shallow parsing of *human now* or *how to start return over* is enough to get the intent of the queries. In the first case, the customer asks for live help; in the second, the customer requires instructions to file from scratch. The utterances are direct requests. However, with *why is my refund low? it was higher last year.*, comparisons between states *current year – last* *year* and *low – high* point to an indirect request for a customer-specific explanation of the current-year refund. Deeper NLU is needed to surface the request. Overall, the idea of using deep NLU has high potential impact, considering that about 75% of customer queries contain features that warrant some level of deep understanding according to the training data distribution (Table 3).

We are exploring a chatbot design wherein incoming queries are routed to either a shallow NLU module that is intent-based or to a NLU that performs semantic analysis. After deep analysis, language feature aggregation weights are used to reroute queries either to intent classification so as to leverage existing answers or to directly live help to improve the conversational flow and customer experience.

References

1. Adam, M., Wessel, M., Benlian, A.: AI-based chatbots in customer service and their effects on user compliance, In: Electron Markets 31, pp. 427–445, `https://doi.org/10.1007/s12525-020-00414-7` (2021)
2. Adamopoulou, E., Moussiades, L.,: Chatbots: History, technology, and applications, In: Machine Learning with Applications, Volume 2 (2020)
3. Brandtzaeg, P., Følstad, A.: Chatbots: changing user needs and motivations. Interactions. 25. pp. 38-43. (2018). 10.1145/3236669.
4. Filipczyk, B., Gołuchowski, J., Paliszkiewicz, J., Janas, A.: Success and Failure in Improvement of Knowledge Delivery to Customers using Chatbot — Result of a Case Study in a Polish SME, In: Successes and Failures of Knowledge Management, pp. 175-189. (2016)
5. Følstad A., Skjuve M., Brandtzaeg P.B.:Different Chatbots for Different Purposes: Towards a Typology of Chatbots to Understand Interaction Design, In: Bodrunova S. et al. (eds) Internet Science. INSCI 2018. Lecture Notes in Computer Science, vol 11551. Springer, Cham. (2018) `https://doi.org/10.1007/978-3-030-17705-8_13`
6. Galitsky, B.: Chatbot Components and Architectures. In: Developing Enterprise Chatbots. Springer, Cham (2019)
7. Janssen, A., Rodríguez Cardona, D., Breitner, M.H.: More than FAQ! Chatbot Taxonomy for Business-to-Business Customer Services. In: Chatbot Research and Design. CONVERSATIONS 2020. Lecture Notes in Computer Science, vol 12604. Springer, Cham. (2020)
8. Janssen, A., Grützner, L., Breitner, M.: Why do Chatbots fail? A Critical Success Factors Analysis, In: International Conference on Information Systems (ICIS), Austin, Texas, (2021)
9. McShane, M., Nirenburg, S.: Linguistics for the Age of AI. 1st edn. 448 pp. The MIT Press, Massachusetts (2021), `https://doi.org/10.7551/mitpress/13618.001.0001`
10. Seeger, A.-M., Heinzl, A.: Chatbots often Fail Can Anthropomorphic Design Mitigate Trust Loss in Conversational Agents for Customer Service?, In: Proceedings of the European Conference on Information Systems, EICS, Morocco (2021)