Exploiting weak-supervision for classifying Non-Sentential Utterances in Mandarin Conversations

Xin-Yi Chen, Laurent Prevot

Exploiting weak-supervision for classifying Non-Sentential Utterances in Mandarin Conversations

Xin-Yi Chen
The Hong Kong Polytechnic University
Department of Chinese and Bilingual Studies
xysimba.chen@connect.polyu.hk

Laurent Prévot
Aix Marseille Université & CNRS
Laboratoire Parole et Langage
laurent.prevot@univ-amu.fr

Abstract

Non-sentential or fragmentary utterances (NSU) constitute a significant part of the productions in a conversation. Although seemingly incomplete in form, they convey full pragmatic meaning in the context. In the past, their classification had been approached with supervised methods (Fernández et al., 2007; Wong, 2018). Such approaches require relatively large annotated data sets. We explore an approach (Ratner et al., 2017a) that allows the reduce significantly the amount of annotated data needed thanks to strategic use of linguistic knowledge. We explore this method for classifying NSUs in Mandarin conversation corpus. Our evaluation shows that promising results can be obtained with a minimal amount of annotated training data.

1 Introduction

In dialogue, besides well-formed complete sentences, a sizeable amount of utterances are fragments that could be understood without a problem in their context. Traditional grammar attends mainly to written texts and canonical sentence analysis. The oral language has been often regarded as bad, spontaneous, and wrong, in summary not an appropriate research object, as (Blanche-Benveniste, 1997) regrets it. But as interest in oral communication gets more attention, terms like "fragments", "Nonsententials" in (Barton, 1991) or "Non Sentential Utterances" (hereafter NSU) in (Fernández et al., 2007) have also attracted more investigation.

The expressions in example 1 below may sound familiar.

(1) What now?
Not you.
What's for supper? - Ground Beef Tacos.

Even though they are generally short, such utterances constitute an active part of the conversation. They contribute to the efficiency of the conversation flow. The interpretation of NSU is essential for linguistic theories that attempt to get serious about language as it is produced in its most natural and pervasive setting, and also for applications, like dialogue systems. It can be done in different ways, as discussed in (Ginzburg, 2012, p:229). The analysis result can be implemented in human-machine dialogue systems in various domains such as client service or computer aided language teacher.

The percentage of NSU among other utterances in conversation corpus is non negligible, 11.15 % in (Fernández and Ginzburg, 2002), 9% in (Fernández et al., 2007), 10.2 % in (Schlangen and Lascarides, 2003). We think the study of NSUs is useful because of the high frequency mentioned above. What's more, the understanding of NSUs and their classification from their context is not always easy. Even a simple "what" can express various emotions and can have different functions in a context. Apart from the most common function as plain question, it can also express Happiness, Surprise, Sadness, Anger, Disgust or even Fear.

Second, the definition of NSUs can have an impact on the classification of NSU, the inclusion and exclusion of categories can be flexible according to the theories and purpose of classification, the classification criteria could be syntactic leading, semantic

leading or a mix of standards. The treatment of some fragments like *'Greetings'* and *'Filler'* can make a difference in the counts. We will see the detailed discussion in section 2.

The paper is structured as follows. Section 2 presents the related work of utterance classification, including Dialogue Acts and Non-Sentential Utterances. Section 3 introduces the data and methodology. Section 4 provides a qualitative and quantitative description of our corpus and the results of the manual labeling. Section 5 summarizes the labeling functions used in this article. Section 6 talks about the modelling and classification experiment in our work. Section 7 is about the evaluation of the model. Section 8 concludes the article.

2 Related Work

2.1 Non-Sentential Utterances

The NSU taxonomy proposed in (Fernández and Ginzburg, 2002) is supposed to be the first "comprehensive, theoretically grounded classifications of NSU in large-scale corpus". The classification is based on work grounded in British National Corpus (BNC), the classification take into consideration both a relatively complex syntax and the context dynamics. In (Fernández et al., 2007), several machine learning experiments were carried out to get an optimal classification result. The features selected for machine learning in this article is limited in a few "meaningful" ones instead of many arbitrary ones. The features selected for NSU classification came from (i) the utterance itself, (ii) its antecedent, and (iii) their relationship. It results in three sets of features in total: *NSU features, Antecedent features* and *Similarity features*. The NSU features include four aspects, whether it is proposition or question, presence of wh-word, yes/no word, and different lexical items. The antecedent features are similar to those of NSU features, but it also looks at whether it is a finished utterance. The similarity features is a comparison of the utterance and its antecedent, mainly about the repeated words and POS tags and their proportion. Another machine learning experimentation work for classification of NSU is based on the work of (Fernández et al., 2007) with more advanced features in (Dragone, 2015).

The taxonomy can be adapted for languages besides English, following the work of (Fernández et al., 2007), the work of (Wong and Ginzburg, 2013) in classifying NSUs in Chinese adds seven subcategories because of the particular behavior of modal verbs in Chinese. The classification we choose is (Wong, 2018), which is based on the work of (Fernández et al., 2007) in adding some classes considering particular behaviors in Chinese Mandarin with extended discussion of each category compared with (Wong and Ginzburg, 2013).

2.2 Utterance Classification

NSU classification is an utterance classification task, of the same kind as the better known Dialogue Act tagging (Stolcke et al., 2000). Dialogue Act (DA) is about the meaning at the illocutionary level defined in (Austin, 1962), which is the intent or effect produced along with the things being said. In (Stolcke et al., 2000), it is said that DAs can be considered "as a tag set that classifies utterances according to a combination of pragmatic, semantic, and syntactic criteria." The DA labels demonstrate the hidden information of the utterance for higher-level processing. It can be used in the interpretation and generation and prediction of utterances and their functions in dialogue systems, as stated in (Stolcke et al., 2000). Therefore DA-tagging is a major applicative task for NLP and Human-Machine Interaction.

Lexical and prosodic cues are both useful for the dialogue act classification. It is observed that some words are symbolic of some DAs. For example, in (Stolcke et al., 2000), "92.4% of the uh-huh's occur in Backchannels, and 88.4% of the trigrams '(start) do you' occur in Yes-No-Questions." For some shared patterns, the differentiation is by pronunciation.

The methodology in DA classification bears similarity with NSU classification. Nevertheless, DA and NSU have differences in their theoretical frameworks and distinctions in aspects such as label uniqueness. NSU is an utterance that is not realized by a full syntactic sentence but produces an effect just like sentential utterances. All utterances can receive a DA label, but only those fragments with incomplete syntactic structure and full semantic value can be labeled as NSU.

By many aspects such as their size as well as their lack of completeness, NSUs can be confused with *disfluencies*. Shriberg (Shriberg, 1996) talked about several types of disfluency: *filled pause, repetition, substitution, insertion, deletion and speech error*. In (Tseng, 1999)'s exploration of modeling the disfluency, there are features found to be useful in the detection of disfluency on the syntactic side: the linguistic length, the syntactic category, the construction types, the location of interruption, the repair onset, and the repair offset. These features could be useful in our examination of disfluency in our corpus. In (Tseng, 2003) 's research about repairs and repetitions in spontaneous Mandarin, the editing term (an indication of speech repair such as "well" "I mean" or filled pauses) is found to be useful in the detection of repetition and repairs.

3 Methodology

A large quantity of training data is necessary for machine learning tasks. But labeled data are not easy to get. Snorkel (Ratner et al., 2017b) provides a solution to this bottleneck by using labeling functions to generate a large amount of labeled data. As stated in (Ratner et al., 2017b), based on theories and experiments, Snorkel has proven effective in training high-accuracy machine learning models, even using potentially lower-accuracy inputs. It has been recently applied to high-level NLP such as discourse parsing in (Badene et al., 2019).

Weakly supervised tools like Snorkel allows for quickly labeling extensive data with minimal but expert manual involvement. The use of Snorkel is to write some labeling functions (LF) to produce some useful training data with labels. A labeling function is a rule that attributes a label for some subset of the training data set. Using Snorkel, it will train a model that combines all the rules defined written to estimate their accuracy, along with the overlaps and conflicts among different labeling functions.

The workflow of Snorkel distinguishes from traditional machine learning approaches; it is based on a data programming paradigm. Briefly, it is composed of two phases, and the first is to produce estimated labels using a generative model, the second is using these labels to train the ultimate model, a discriminative model.

Within this design philosophy, the system design of Snorkel can be divided into three phases: first, pre-processing of the data to have the reorganized data for later use, such as word segmentation and POS tagging. Second, writing labeling functions. Labeling functions do not need to be entirely accurate or exhaustive and can be correlated. Snorkel will automatically estimate their accuracies and correlations in a provably consistent way, as introduced in (Ratner et al., 2016). Third, after the evaluation and calibration of the LFs, we decide on an optimal set of LFs to produce a set of labels to train a model.

4 Data and Manual labelling

Extensive conversational data are limited in numbers. We are interested in the real-time conversational data in talking form transcribed in textual form instead of texts generated in instant-messaging tools. The data we used in this study is from LDC's CALLHOME Mandarin Chinese collection. This is a telephonic conversation corpus, with audio files and transcriptions. The language was in Mandarin even though the participants are from different provinces of China. The corpus includes 120 transcripts in total, and each is a five or ten-minute segment from the telephone speech files. From the description on the website [1], the transcripts are already tokenized automatically using a tool called the Chinese Lexical Analysis System (ICTCLAS). The results were further corrected manually.

In this paper, the corpus concerned is already segmented. In long sentences, an NSU component may appear in the middle, but we won't label it as NSU if it doesn't stand independently. Suppose we deal with a raw corpus not segmented yet. In that case, we will decide the utterance boundary first based on our research question(s) and the conversation context, including syntax, prosody, and pragmatic effect. However, it's also possible to define NSU and describe it first and then extract them or locate them in the corpus.

The original data includes the start time and end time of every segment, the speaker, the textual content of the utterance. In the text transcription, there are also examples of annotation as enrichment of information as illustrated in example 2.

[1] https://catalog.ldc.upenn.edu/LDC2008T17

(2) Examples of annotation [2]
{text}: sound made by the talker. e.g. {laugh} {breath_noise}
//text//: aside (talker addressing someone in background) e.g.// 来 说 ⟨English_Hello,⟩ 您好 . (Come say Hello, hello).//

We processed the data and transformed it into Pandas DataFrame (McKinney, 2010) in order to manipulate it into Jupyter Notebooks (Pérez and Granger, 2007). They are transformed as a table, and the information is divided by columns. The original information is separated into four columns: *Start time*, *End time*, *Speaker*, and *Text transcription*. Based on these, we added other columns (illustrated in figure 1):

- Conversation code: the original code of the file
- Duration : how long the utterance lasts
- Same Speaker: if the utterance is produced by the speaker of the previous utterance (BOOLEAN)
- Latency : a gap between two turns, the Start time minus the previous End time (we only consider the positive value cases)
- Overlap : the duration when more than one person speaks (we only consider the positive value cases)
- Word count: How many units are there in the utterance (depending on the segmentation method, one unit may not necessary correspond to one Chinese character, and the punctuation can be included as well)
- Tagged: the POS tagged text used in this study is attributed by the tool Zpar (Zhang and Clark, 2011)

We have 33485 utterances in total in combining 120 files. Combined with the tagged results, we omit the ones untagged, so we deal with 33431 utterances (229 412 tokens).

[2] http://shachi.org/resources/661

We selected around 5% of the whole data as a sample to tag manually to know the difference between data with NSU tags and the complete data. Only one annotator does the manual annotation for convenience and cost. Then we have another annotator to annotate 7% of the sample data (0.35% of the whole data) to compare with the first annotator's result. We get a kappa score of 0.54 for all the NSU categories and a kappa score of 0.57 for the four most frequent NSU classes((PLAIN ACKNOWLEDGMENT, REPEATED ACKNOWLEDGEMENT, CHECK QUESTION, and INTERJECTION), a kappa score of 0.58 for the four first-level NSU classes (ACKNOWLEDGMENT,QUESTION, ANSWER and COMPLEMENT).

Through a qualitative analysis of the NSU categories in our corpus, we made some adjustments of the classification in (Wong, 2018), the results are shown in table 1.

5 Labelling Functions

When writing labeling functions, there are several strategies: keyword matches, regular expressions, arbitrary heuristics, and third-party models.

In our case, we use the first two strategies combined with three types of cues: the Textual cues, the Timing cues, and the Contextual cues. For each type of signal, we look at the relevant features. There are two variables in (Schlangen, 2005), the structural features and the lexical/utterance-based features. In (Fernández et al., 2007), as mentioned in section 2, there are three sets of features: *NSU features, Antecedent features, and Similarity features*. In (Dragone, 2015), the baseline feature set is the same as in (Fernández et al., 2007), but with extended features at different levels: POS tags, phrase-level, dependency features, turn-taking features, and similarity features. We have chosen these features as presented in the figure 2 based on the characteristics and available information of our corpus.

In our case, the features are used in the writing of labeling functions. Based on the result of LF performance, which is undoubtedly influenced by the majority's classes, the more frequently used features are the keywords, such as feedback/ backchannel word, followed by wh-question word and ques-

Latency	Overlap	Same_speaker	Speaker	Start	Text	Word_count	Tagged
NaN	NaN	False	A	183.47	你 很 爽 哈 你 , 哈 ?	6	你_PN 很_AD 爽_AD 哈_VV 你_PN ,_PU 哈_VV ?_PU
NaN	0.27	False	B	184.93	要 不要 跟 妈 讲话 ?	5	要_VV 不_AD 要_VV 跟_P 妈_NN 讲话 _VV ?_PU
0.08	0.00	False	A	186.03	啊 ?	1	啊_VA ?_PU

Figure 1: Head of the pre-processed corpus dataframe

tion final particles. Features used to detect the Sentential Utterances and Disfluency also have good performance. Some features may be not so effective because of some shared words among different NSUs, thus less frequent due to major classes' existence. For instance, "嗯"(um) is typical in PLAIN ACKNOWLEDGEMENT. Still, it can also appear in INTERJECTION or questions, so that we may need a combination of features such as POS tag features and other corpus-related cues.

Our labeling functions can be divided into three types: Keyword-based LF combined with size-related LF, POS tagging LF, Context-related LF. For the three classification models, we set the size-related limitation such as counted words and we used frequent words for each NSU category in the LF, and also frequent POS tag or tag combination, such as demonstrated in figure 3. We also compare the number or promotion of shared patterns between the utterance and its precedent. For the SU class, we also have LF targeting at disfluency with size-related LF, such as Duration and Word_count, contextual cues (two consecutive utterances produced by the same speaker) and POS tag cues.

6 Modelling and Classification

Our goal is to build a model to classify all the NSU classes, we also build two extra classification models for comparison, one with the four first-level classes ACKNOWLEDGMENT,QUESTION, ANSWER and COMPLEMENT, and another with the four most frequent NSU classes(PLAIN ACKNOWLEDGMENT, REPEATED ACKNOWLEDGEMENT, CHECK QUESTION, and INTERJECTION).

It should be noted the final set for each model only includes the LFs without serious incorrectness. Otherwise, it will only harm the model so that if an LF has more incorrect than the correct cases, we tend to exclude them, especially when the ratio is significant. Based on the result and after the error analysis, this problem could not be solved; we do not have LFs for each NSU. For the main classes model, we didn't get a proper LF for the class REPEATED ACKNOWLEDGEMENT, for the first-level classification model, the ANSWER class, and COMPLEMENT class LF don not enter in the final set. For all-class model, only PLAIN ACKNOWLEDGMENT, CHECK QUESTION, and INTERJECTION) entered in the final set.

The results are presented in table 3. For each model, we run the experiment in three conditions:

- Baseline: with the definite majority class acknowledgment (the most frequent one) with 58% in our sample data frequency;
- System: with all the classes in each model, no use of punctuation (the training label difference in these three conditions can be seen in table 2);

	NSU Class
	A. Acknowledgement
1	Plain Acknowledgement
2	Repeated Acknowledgement
3*	Verbal Acknowledgement
4*	Helpful Acknowledgement
5*	Re-Affirmation
	B. Questions
6	Clarification Ellipsis
7	Sluice
8*	Nominal Predication
9	Check Question
	C. Answers
10	Short Answer
11	Affirmative Answer
12	Repeated Affirmative Answer
13*	Verbal Affirmative Answer
14*	Helpful Affirmative Answer
15	Rejection
16*	Verbal Rejection
17	Helpful Rejection
	D. Complement
18	Filler
19*	Correction
20*	Interjection
21	Propositional Modifier
22	Factive Modifier
23	Bare Modifier Phrase
24	Conjunction + Fragment

Table 1: Classification of NSU in (Wong, 2018)

Feature	Description
NSU feature	Presence of wh-question word
	Presence of question final particle
	Presence of propositional modifier word
	Presence of feedback/backchannel word
	Presence of interjection
	Presene of factual modifier word
	Presence of modal word
	Presence of polar particle
	Presence of heavy tags (noun, verb, ajective and adverb)
	Presence of disfluency
	Presence of repeated pattern(word/tag) in the utterance
	Is the utterance a question or not?
Antecedent features	Presence of wh-question word
	Presence of question final particle
	Presence of disfluency
Structural features	Is the two consecutive utterance produced by the same speaker?
	Common pattern(word/tag) between the utterance and its precedent

Figure 2: Features and description

- Topline: also with all the classes in each model, including punctuation (provided by the transcript) as cues.

Snorkel's Label Model can learn the dependency among the LFs, and its output is an array of single probabilistic training labels. As explained in (Ratner et al., 2016), there are four types of dependency among the LFs: "similar, fixing, reinforcing, and exclusive." A dependency graph will be calculated and established. Overall, the model will give a more data-balanced decision for the data points where there are conflicting LFs.

The Majority Label Voter of Snorkel takes the majority vote for each data point; each LF will cover a portion of data. Its inadequacy is that each vote of the LF are considered of equal efficiency, but this is not the case. Snorkel's Label Model deals with the correlation among LFs when combining all the outputs of the LFs.

As we have mentioned the workflow of Snorkel in section 3, Snorkel's Label Model's output is then used to train the ultimate discriminative model, such as a Scikit-Learn classifier.

The Label Model Accuracy is not always higher than the Majority Vote Accuracy. In (Ratner et al., 2017b), it's explained that for very sparse label matrices (almost no conflicts among LFs) or very dense label matrices (a lot of conflicts among LFs) will probably lead to this result. The F1 score is a Micro average for the multiclass setting, that "calculates metrics globally across classes, by counting the total true positives, false negatives and false positives", as explained in (Sasaki, 2007).

7 Evaluation

So the result of a task to detect just the majority class ACKNOWLEDGMENT from the Sentential Utterace (SU) and the rest of the NSU classes is acceptable, but the abstain votes from the other NSU classes can explain the gaps with the system condition. The small difference among all these three conditions can be attributed to the outcome of the final labeling function sets. Because we omit the LFs with apparent imprecision, we are left with an LF set targeting classes for some major classes like ACKNOWLEDGMENT and a few effective others for the rest.

```
@labeling_function()
def ackplain_pos(x):
    tags = [utt.split('_')[1] for utt in x['Tagged'].split()]
    if x["Word_count"] < 2:
        for tag in tags:
            if tag not in ['NN','AD','VA','PU']
                return ABSTAIN
    return ACKPLAIN
```

Figure 3: Example of LF using unigram POS cues, PLAIN ACKNOWLEDGEMENT

Transcript	Baseline	System	Topline
呃哼/ Uh-huh	SU	Interjection	Interjection
寄出来了/ It's coming out	SU	Repeated Ack.	Repeated Ack.
好不好? / All right ?	SU	Check Question	Check Question

Table 2: Comparison of training labels in baseline, system and topline situations in Majority class classification model

		Baseline	System	Top-line
All-class classification model	Majority Vote Accuracy	72.70%	72.70%	73.70%
	Label Model Accuracy	72%	72%	75.30%
	F1 micro	0.75	0.78	0.82
	Scikit-learn classifier test accuracy	68.70%	74.30%	75.70%
First-level classification model	Majority Vote Accuracy	80.00%	79.70%	84.00%
	Label Model Accuracy	79%	77%	84%
	F1 micro	0.79	0.8	0.83
	Scikit-learn classifier test accuracy	74.30%	74.00%	76.00%
Majority class classification model	Majority Vote Accuracy	67.30%	74.30%	75.00%
	Label Model Accuracy	67%	74%	74.30%
	F1 micro	0.77	0.82	0.77
	Scikit-learn classifier test accuracy	67.70%	73.30%	74.70%

Table 3: Performance of three models in three conditions

The all-class classification model's performance can be attributed to the number of classes and the affiliation relation between them. The 24 tags are mutually exclusive, but some can be grouped under a first-level category. Besides, the SU class is the opposite of all the other classes. With its relatively high frequency, in a binary situation, when we only need to distinguish SU and NSU. Still, in our multi-class setting, one class's negative classification is not yet realized in Snorkel. For some classes, even though we have posed some limits on the counted word number and duration, the LF still targets many SU (including disfluency cases). Consequently, there are many false-positives for some LFs, especially for some minority categories, such as for different sub-categories under ANSWER. Extremely unbalanced data as reference, they do not have a single case present in the labeled data set.

Also, the similarity between ANSWER and ACKNOWLEDGEMENT makes it hard to classify the ANSWER and its sub-classes. They have shared words and sometimes similar scope of counted words; the most credible way is by solving whether the previous utterance is a question. But when we do without the punctuation, the performance is not so good, neither. Some INTERJECTION words are also confused with the ACKNOWLEDGEMENT.

It's exceptionally delicate when dealing with some classed heavily depending on the semantic relationship. For the all-class model, we haven't come

up with LFs for BARE MODIFIER PHRASE and CORRECTION who are hard to capture.

8 Conclusion

In this article, we present our work regarding non-sentential utterances automatic classification. NSUs are utterances partial syntactically but convey integral meaning semantically. We chose one classification for Chinese Mandarin and test it with a telephone conversation corpus, using a weak supervision method to build a model for automatic labeling.

From a broader perspective, the approach adopted shows interesting results. It constitutes an efficient way to combine domain experts (here linguists) with state-of-the art machine learning techniques.

Future development For classes with barely any coverage in the reference data set, such as the subcategories in ANSWER, we can put more data of these classes for the model training and use some data augmentation method so that we can test and find the LF for these classes.

Dealing with classes easily confused with majority class, such as INTERJECTION and ACKNOWLEDGEMENT, we may need audio-related information to distinguish them, such as intensity and energy of utterances. Prosodic information has appeal in separating question from declarative with the rising tone at the end for the Mandarin.

To find the semantic connection for a particular utterance in cases, especially when there are no repeated patterns, we need tools to present the relatedness not only for two consecutive utterances but with a flexible contextual window.

Acknowledgments

We would like to thank Pierre Magistry for helping with the POS-tagging as well as anonymous reviewers for extremely valuable comments. All remaining errors are ours.

References

[Austin1962] John Langshaw Austin. 1962. *How to do things with words*. William James Lectures. Oxford University Press.

[Badene et al.2019] Sonia Badene, Kate Thompson, Jean-Pierre Lorré, and Nicholas Asher. 2019. Data programming for learning discourse structure. In *Association for Computational LInguistics (ACL)*.

[Barton1991] Ellen Barton. 1991. Nonsentential Constituents and Theories of Phrase Structure. In Katherine Leffel and Denis Bouchard, editors, *Views on Phrase Structure*, pages 193–214. Springer Netherlands, Dordrecht.

[Blanche-Benveniste1997] Claire Blanche-Benveniste. 1997. *Approches de la langue parlée en français*. Collection L'essentiel français. Ophrys, Gap Paris. graph. 21 cm. Bibliogr. p. 149-151. Index.

[Dragone2015] Paolo Dragone. 2015. Non-sentential utterances in dialogue: Experiments in classification and interpretation. *CoRR*, abs/1511.06995.

[Fernández and Ginzburg2002] Raquel Fernández and Jonathan Ginzburg. 2002. Non-sentential utterances in dialogue: A corpus-based study. In *Proceedings of the Third SIGdial Workshop on Discourse and Dialogue*, pages 15–26, Philadelphia, Pennsylvania, USA, July. Association for Computational Linguistics.

[Fernández et al.2007] Raquel Fernández, Jonathan Ginzburg, and Shalom Lappin. 2007. Classifying Non-Sentential Utterances in Dialogue: A Machine Learning Approach. *Computational Linguistics*, 33(3):397–427, September.

[Ginzburg2012] Jonathan Ginzburg. 2012. *The Interactive Stance: Meaning for Conversation*. Oxford University Press UK.

[McKinney2010] Wes McKinney. 2010. Data structures for statistical computing in python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 51 – 56.

[Pérez and Granger2007] Fernando Pérez and Brian E. Granger. 2007. IPython: a system for interactive scientific computing. *Computing in Science and Engineering*, 9(3):21–29, May.

[Ratner et al.2016] Alexander Ratner, Christopher De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. 2016. Data programming: Creating large training sets, quickly. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 3574–3582, Red Hook, NY, USA. Curran Associates Inc.

[Ratner et al.2017a] Alexander Ratner, Stephen H. Bach, Henry R. Ehrenberg, Jason Alan Fries, Sen Wu, and Christopher Ré. 2017a. Snorkel: Rapid training data creation with weak supervision. *CoRR*, abs/1711.10160.

[Ratner et al.2017b] Alexander J. Ratner, Stephen H. Bach, Henry R. Ehrenberg, and Chris Ré. 2017b. Snorkel: Fast Training Set Generation for Information

Extraction. In *Proceedings of the 2017 ACM International Conference on Management of Data - SIGMOD '17*, pages 1683–1686, Chicago, Illinois, USA. ACM Press.

[Sasaki2007] Yutaka Sasaki. 2007. The truth of the F-measure. page 5, October.

[Schlangen and Lascarides2003] David Schlangen and Alex Lascarides. 2003. The interpretation of non-sentential utterances in dialogue. page 10.

[Schlangen2005] David Schlangen. 2005. Towards finding and fixing Fragments—Using ML to identify non-sentential utterances and their antecedents in multi-party dialogue. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 247–254, Ann Arbor, Michigan, June. Association for Computational Linguistics.

[Shriberg1996] Elizabeth Shriberg. 1996. Disfluencies in switchboard. In *Proceedings of the International Conference on Spoken Language Processing*, pages 11–14, Philadelphia, PA, October.

[Stolcke et al.2000] Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–374.

[Tseng1999] Shu-Chuan Tseng. 1999. *Grammar, Prosody and Speech Disfluencies in Spoken Dialogues*. Ph.D. thesis, University of Bielefeld.

[Tseng2003] Shu-Chuan Tseng. 2003. Repairs and repetitions in spontaneous mandarin. In *ISCA Tutorial and Research Workshop on Disfluency in Spontaneous Speech*.

[Wong and Ginzburg2013] Kwong-Cheong Wong and Jonathan Ginzburg. 2013. Investigating non-sentential utterances in a spoken chinese corpus.

[Wong2018] Kwong-Cheong Wong. 2018. *Classifying Conversations*. Ph.D. thesis, Université Paris Diderot - Paris 7.

[Zhang and Clark2011] Yue Zhang and Stephen Clark. 2011. Syntactic processing using the generalized perceptron and beam search. *Computational Linguistics*, 37(1):105–151.