The Thirty-Second International Florida
Artificial Intelligence Research Society Conference (FLAIRS-32)

Using Linguistic Context to Learn Folksonomies from Task-Oriented Dialogues

Gregory Moro Puppi Wanderley, Emerson Cabrera Paraiso
Graduate Program in Informatics
Pontifícia Universidade Católica do Paraná
Rua Imaculada Conceição, 1155
Curitiba, Brazil 80215-901
{gregory, paraiso}@ppgia.pucpr.br

Abstract

Dialogue systems intend to facilitate the interaction between humans and computers. A key element in a dialogue system is the conceptual model which represents a domain. Folksonomies are very simple forms of knowledge representation which may be used to specify the conceptual model. However, folksonomies by nature have ambiguity. In this paper, we present a method which uses linguistic context for learning folksonomies from task-oriented dialogues. The linguistic context can be useful for reducing ambiguity, for instance, when using the folksonomies for interpreting utterances. Experiments show that the learned folksonomies increase the accuracy of the interpretation compared when not using the contextual information.

Introduction

Currently, the booming interest in chatbots and conversational interfaces for facilitating the interaction between humans and computers have emphasized the importance of dialogue systems. Aiming to coordinate such interaction, dialogue systems can simplify substantially a number of daily tasks such as in call centers, online shopping or consulting services (Burtsev et al. 2018).

To interpret utterances, dialogue systems make use of different stages of Natural Language Processing (NLP) such as morphological, syntactic, semantic and pragmatic analysis (Jurafsky and Martin 2014). Moreover, a key component for supporting the interpretation process is the interaction of all these stages with the conceptual model which represents and describes the domain of the dialogue system.

Recently, we have proposed the FolksDialogue method which specifies the conceptual model through folksonomies, learning them automatically from task-oriented dialogues (Wanderley et al. 2015). Folksonomies are forms of knowledge representation that emerge from the tagging process in collaborative tagging systems (Peters 2009). The tagging process corresponds to the assignment of tags to resources by common users. Resources can be any object that users are interested in tag such as photos or videos, aiming to describe or classify them.

Comparing to more complex knowledge structures such as ontologies which are usually time consuming and costly to be obtained, mainly because of the consensus needed to be reached between domain experts, folksonomies are simpler to implement and use (Echarte et al. 2007). However, folksonomies carry by nature issues related to ambiguity. That is, the tags are freely chosen by users using their own vocabulary, thus being subject to issues such as the use of homonyms (Xu et al. 2018) that may affect, for instance, the interpretation of utterances.

In this paper, we present the C-FolksDialogue method (C for context) which is an evolution of our past approach, for trying to reduce ambiguity when interpreting utterances. The C-FolksDialogue uses linguistic context for learning folksonomies automatically from task-oriented dialogues. The linguistic context consists of linguistic features from the learning dialogues such as dependency relations and Part of Speech to try to contextualize better the folksonomy tags. Moreover, another novelty of the C-FolksDialogue is to evolve existing dialogue folksonomies, instead of every time build them from scratch.

We performed an experiment to verify if the folksonomies learned with the C-FolksDialogue reduce ambiguity when interpreting utterances, verifying whether or not they belong to the domain represented by the folksonomy. In this paper, first we survey related work. Then, we provide a formal background of folksonomies learned from dialogues. After that, we present our C-FolksDialogue method. Next, we show experiments and discuss results. Finally, we end with conclusions and directions for future work.

Related Work

The focus of this section is to present works which explicitly show that the structure they obtain from collaborative tagging systems is a folksonomy.

The authors in (Plangprasopchok and Lerman 2009) proposed constructing folksonomies by aggregating personal hierarchies (collections on Flickr). The final structure generated by the union of all personal hierarchies, and which represents the folksonomy was a tree of tags. The technique used for obtaining the folksonomy was based on relational clustering with similarity measures.

Strohmaier et al. implemented and evaluated three classes of algorithms for the induction of folksonomies. The three algorithms created hierarchical structures of tags. The process of obtaining the folksonomies by the algorithms, involved methods of clustering applied recursively and the use of the hierarchical K-Means algorithm to data from systems like Flickr and Last.fm.

To the best of our knowledge, our previous work (Wanderley et al. 2015) is currently the only one taking advantage of the social dimension that folksonomies and dialogues have in common, and performs the learning of these knowledge structures from such dialogues. Moreover, it is also the unique approach to regard folksonomies as a knowledge representation of a given domain, which interpret utterances indicating whether they belong or not to this domain. However, the folksonomies learned in this work used our past FolksDialogue method (mentioned briefly in the Introduction) which did not take into account an explicit context, useful, for instance, for reducing ambiguity and improving the interpretation of utterances.

Folksonomies Learned from Dialogues

Folksonomies can be defined through a formal and well accepted tripartide model in which they are composed by three entities: users, tags and resources. Based on the approach proposed by Schmitz et al., a folksonomy can be defined as a tuple $F := (U, T, R, Y)$, where U, T, R are the finite sets of users, tags and resources, respectively, and Y is the ternary relation between them called "tag assignement." The "personomy" P_u of some user $u \in U$ corresponds to the set of all tag assignments that she has generated in the tagging process of a given domain. Thus, we can infer that a folksonomy is the union of all personomies of all users. Moreover, any two tags t_a, t_b often appearing together tagging the same resources may have a relationship b between them, if and only if they have appeared together at least x times. Note that, x can be considered as a weight in this relationship.

A common feature between folksonomies and dialogues is their social dimension. In task-oriented dialogues, on of the main characteristics is the existence of two types of interlocutors, one asking for help (we call user) and the other with knowledge of the domain (we call attendant), aiming at joint completion of a specific task (Artstein et al. 2017) like in a call center. Through that, we extend the folksonomy definition for learning it from dialogues (Wanderley et al. 2015) and represent the entities users, tags and resources as follows: users are the attendants of task-oriented dialogues, tags are the nouns of the utterances produced by the attendants, and finally, resources are the own utterances of the attendants. The reason for using the attendants as the entity users, and their utterances as the entity resources is that we assume they have full knowledge of a given domain. By contrast, interlocutors of the type user are who need help to solve some task. Thus, they can state anything in their utterances, even whether those things are outside and divergent of a given domain.

Definition 1. A subset of users l belongs to a given attendant a and is composed of all users with whom she has dialogued in a given domain. Each attendant has one, and only one, subset of users. Formally, let

- A be the finite set of attendants (a be an attendant belonging to A);
- U be the finite set of users (u be a user belonging to U);
- Du be a function $Du : A \times D \rightarrow U$ that returns the user attended by an attendant in some dialogue;
- Ut be the set of utterances of all dialogues.

The subset of users for the attendant a (a is a constant) can then be defined by the predicate $l : \forall d((a, d) \in A \times D) \rightarrow l(Du(a, d))$.

Definition 2. A folksonomy obtained from task-oriented dialogues is defined as a tuple $F := (A, T, R, U, Y')$, where

- A is the finite set of the users of the folksonomy. That is, the attendants of the task-oriented dialogues;
- T is the finite set of tags, which are the nouns of the utterances that attendants have generated in the dialogues;
- R is the finite set of resources of the folksonomy, and consists of the attendants' utterances;
- U is the finite set of users;
- Y' is the quaternary relation among the above, i.e., $Y' \subseteq A \times T \times R \times U$.

Thus, a folksonomy obtained from task-oriented dialogues is represented by a "quadripartite model" that has four dimensions: attendants, tags, resources and subset of users. Moreover, in a dialogue folksonomy, two any tags t_a, t_b will have a relationship $b \in B$ (set of relationships between tags) between them if and only if such tags appear together (tagging the same resources) at least x times. The weight w for the relationship is the number of dialogues in which the relevant tags have appeared together.

Using Linguistic Context to Learn Folksonomies from Task-Oriented Dialogues

In this section we present the C-FolksDialogue method for learning folksonomies with linguistic context from dialogues. First, to better understand our approach we introduce the concept of linguistic context in dialogue folksonomies.

Formal Definition

To reduce ambiguity and define better the meaning of the tags belonging to a dialogue folksonomy, we propose an approach that makes use of a linguistic context. We introduce in following the definition of linguistic context in a folksonomy learned from task-oriented dialogues.

Definition 3. The linguistic context of a tag (noun) t, in a dialogue folksonomy, consists of a list of tuples of the kind: *(dependency element, Part of Speech, dependency relation)*. Each tuple corresponds to a dependency relation that t has in the folksonomy resources (utterances). In the tuple, the "dependency element" is an element with a dependency relation with the tag, the "Part of Speech" is the Part of Speech of the "dependency element," and the "dependency relation" is the

```
<lng-ctx-folk>::= {<lng-ctx-tag>}*
<lng-ctx-tag>::= <tag>{<resource>
                        <lng-ctx-resource>)}*
<lng-ctx-resource>::= {(<dep-element>
            <Part-of-Speech><dep-relation>)}*
<tag>::= <noun>
<resource>::= <utterance>
```

Figure 1: The linguistic context of a folksonomy.

own dependency relation between the tag and the "dependency element." Then, the entire linguistic context of a folksonomy learned from task-oriented dialogues is a list containing all the linguistic contexts of its tags. Formally, the linguistic context of a folksonomy ("lng-ctx-folk") can be defined by the grammar in Backus-Naur form (BNF) shown in Figure 1.

To give an example, suppose that the noun "ticket" is tag and the utterance "I bought your tickets for the movie." is a resource in a folksonomy. Then, the linguistic context of the tag "ticket" could be the represented by the tuple (buy, verb, direct-object). This means that "ticket" has a dependency relation "direct-object" with the verb "buy." Note that, we used the lemma of the verb (i.e., infinitive form) in the tuple. Moreover, one may use a parser to find the required linguistic elements such as the dependency relations.

The C-FolksDialogue Method

The C-FolksDialogue method consists of two activities, named Preprocessing and Learning. To better understand our method, we will describe it from the perspective of evolving a folksonomy. The overall approach consists in taking a new dialogue corpus from the folksonomy domain, and use it to learn a new folksonomy. During the learning activity, the method extracts different elements such as tags and resources from the (old-)folksonomy aimed to evolve. After that, the extracted elements are incorporated in the new folksonomy being learned. Finally, the C-FolksDialogue outputs the learned folksonomy with the incorporated elements, representing then the evolved folkosonomy.

The Preprocessing stage aims to receive as input a new learning dialogue corpus (formally, Dn where d_n is a dialogue belonging to Dn) from to the domain of the evolving folksonomy along with such a folksonomy. The goal is to make them fit for use in the rest of the process. The steps that compose the preprocessing are: "Extract Attendants' Utterances," "Extract Nouns," "Lemmatize," "Remove Duplicate" and "Retrieve Old-Folksonomy Corpus."

The step of "Extract Attendants' Utterances" aims to receive the dialogue corpus Dn and extract only the utterances of the attendants (formally, represented by the set Ut where ut is an attendant utterance). Note that, we assume that the attendants have full knowledge of the domain of the dialogues (section "Folksonomies Learned from Dialogues"). The main purpose of this "filtering" is to forward to the subsequent steps of the method only utterances that represent the relevant domain.

The next step is the "Extract Nouns," which aims to extract the nouns of the utterances from Ut. The goal is to initiate the process of getting the nouns which later will be converted to the tags of the new folksonomy being learned. The identification of the nouns in the utterances of Ut is performed by a morphological analysis through a parser. Formally, the nouns extracted from Ut can be represented by a multiset (which admits repetitions in its elements) S where s is a noun belonging to S.

The step "Lemmatize" intends to lemmatize all the nouns from S through the aid of a lemmatizer. The purpose is to obtain the lemmas of the nouns of S to avoid uppercase or lowercase, and/or singular or plural forms. The output of this step is the lemmatized nouns.

The next step is the "Remove Duplicate" that removes possible duplicates lemmas obtained in the previous step. The output of this step is a list Ls of unique nouns.

Because we are taking the perspective of evolving a folksonomy, the last step of the Preprocessing is the "Retrieve Old-Folksonomy Corpus." The goal of this step is to take as input the "old folksonomy" which is being evolved, and retrieve the dialogue corpus that was used to learn it. Formally, the dialogue corpus of the evolving folksonomy is represented by the set Do, where d_o is a dialogue belonging to Do. Then, the output of the Preprocessing activity is: the list Ls of nouns, the set of attendants' utterances Ut and the corpus Do used to learn the folksonomy to be evolved.

The second activity of the C-FolksDialogue is the Learning. Figure 2 shows the steps of the Learning activity which are: "Obtain Folksonomy Tags," "Obtain Folksonomy Resources," "Extract Old-Folksonomy Tags," "Extract Old-Folksonomy Resources," "Obtain Folksonomy Attendants," "Obtain Folksonomy Users," "Obtain Relationship between Tags," "Build Personomies," "Merge Personomies," "Extract Coreference," "Connect Resources," "Extract Dependency Relations," "Extract Part of Speech," "Add Linguistic Context" and "Synthesize."

The "Obtain Folksonomy Tags" selects nouns from the list Ls (output of the Preprocessing activity) to be the tags of the new folksonomy being learned. To do that, our method ranks the nouns based on their IDF (Inverse Document Frequency) (Sparck Jones 1972) in the dialogues of the corpus Dn along with the corpus Do. The nouns whose have their IDF below a cutoff frequency (Wanderley et al. 2015) are discarded. This is because for this research, the IDF means the importance of each of the nouns in the dialogue corpus, being considered less important nouns outside and diverging the representation of the domain. The nouns that were not discarded are the tags of the set T of folksonomy tags.

After that, because we are evolving a folksonomy, the next steps performed are "Extract Old-Folksonomy Tags" and "Extract Old-Folksonomy Resources." In the former the goal is to retrieve the tags and in the latter the resources of the (old-)folksonomy being evolved, preparing them to be used in the next steps of the learning stage. The input of both steps is the evolving folksonomy, and the output are its sets of tags (T_e) and resources (R_e).

The next step "Obtain Folksonomy Resources" intends to obtain the resources of the folksonomy being learned. Note that the resources of the folksonomies are the utterances of attendants (section "Folksonomies Learned from

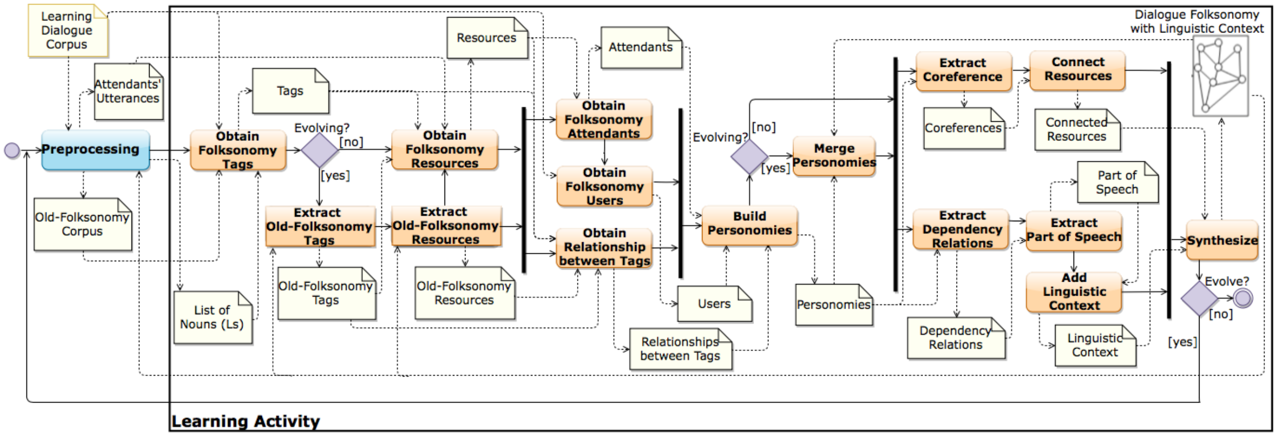

Figure 2: The steps of the Learning activity of the C-FolksDialogue method.

Dialogues"). To select the utterances from the set Ut to become resources, we use the tags of the sets T and T_e. Given the fact that these tags are nouns belonging to the given domain, in this step we verify how many of the nouns of a given utterance are tags. The goal is to verify which of the attendants' utterances belong to the domain to be adopted as resources. For each utterance of the attendants, the method calculates a Ratio of Inclusion p_{ut} (Equation 1) extended from (Wanderley et al. 2015). This ratio measures the percentage of nouns of an utterance that are tags of the folksonomy, i.e., those belonging to the domain. The utterances with their p_{ut} greater than or equal to a cutoff frequency (Wanderley et al. 2015) are adopted as folksonomy resources. The output of this step is the set R of resources.

$$p_{ut} = \frac{|\{s:(s \in ut) \land (s \in T \lor s \in T_e) \land (ut \in Ut)\}|}{|\{s:(s \in ut) \land (ut \in Ut)\}|} \times 100 \quad (1)$$

Next, the step "Obtain Relationships between Tags" finds possible relationships between the tags of the set T, and also between the tags of T with the tags belonging to T_e. First, it generates all possible pairs of tags from the set T, and between T and T_e. Then, for each generated pair of tags we calculate the frequency that the two terms part of the pair appear together tagging the same resources in the sets R and R_e. That is, the pairs of tags with frequencies lower than a cutoff frequency (Wanderley et al. 2015) are discarded, otherwise, they will have a relationship between their tags and will be part of the set B of relationships between tags.

In parallel with the previous step, the C-FolksDialogue performs the step of "Obtain Folksonomy Attendants." The goal of this step is to obtain the set A of attendants of the folksonomy being learned. For each resource of the set R, the method extracts all the different attendants "a" that exist, building the set A of the folksonomy being learned.

After that, the step "Obtain Users" acquires the set U of users of the folksonomy being learned. The set U of interlocutors of type user (asking for the assistance of the attendants) is obtained by extracting all interlocutors of type u from the corpus Dn used as input in the method.

In the next step "Build Personomies" the purpose is to build the personomies of the attendants, i.e., the users of a dialogue folksonomy. From the sets A, T, R, U, and B, obtained in the previous steps, the method performs the connection between the elements of these sets through the quaternary relation Y' (section "Folksonomies Learned from Dialogues"). After that, the tags of the personomies are then connected through relationships in set B (relationships between tags).

Then, because the C-FolksDialogue is evolving a folksonomy, the next step is "Merge Personomies." The goal is to merge the personomies obtained in the "Build Personomies" with the personomies of the (old-)folksonomy being evolved. First, it looks for personomies belonging to a same given attendant a and then, it connects all the elements of the sets T, R and U of such personomies to a single one. When an attendant has directly only a single personomy, for instance, it is a new attendant in the dialogues, her personomy is kept with the others without merging.

After merging, the next steps intend to enrich the folksonomy being evolved with linguistic context . The step "Extract Coreference" aims to verify in the personomies which resources may be related to each other, for instance, share the same context. For all resources (utterances) belonging to the same dialogue (i.e., connected to the same dialogue user u), the method tries to verify if there are coreferences such as pronouns to be resolved between such resources. To do that, the C-FolksDialogue uses a parser for performing the coreference resolution. The output of this step are the personomies along with the coreferences found between the resources.

In the "Connect Resources" step, the goal is to connect the resources of the personomies according to the coreferences extracted in the previous step. The output of this step are the personomies with the relations between the resources.

Next, in parallel with the "Extract Coreference," the method performs the step of "Extract Dependency relations." The purpose is to extract for each tag in the personomies, all the dependency relations (e.g., nominal subject or adverbial modifier) they have with other linguistic elements (e.g., nouns or verbs) in the resources (utterances) they are connected. To do that, our method uses a dependency parser for extracting relations between the tags (nouns) and the other linguistic elements they have dependencies in the resources.

Then, the "Extract Part of Speech" step aims to extract the Part of Speech (e.g., noun, verb or adverb) of the linguistic elements that have a dependency relation with the tags. To identify the Part of Speech, the C-FolksDialogue uses a parser for performing morphological analysis in the linguistic elements. The output of this step is the Part of Speech of the linguistic elements with a dependency relation with the tags.

Now, the method has all the elements required for adding the linguistic context to the evolving folksonomy, i.e., elements showing dependencies with tags, their Part of Speech, and their dependency relations (section "Formal Definition"). In the step "Add Linguistic Context," the goal is to add the linguistic context of the tags to the relations between them and the resources in the personomies. This means that a given tag t connected to a resource r can be contextualized "linguistically" in r through the linguistic context in such a relation.

The last step of the C-FolksDialogue method is "Synthesize" which aims to synthesize in the personomies the relations between resources with the linguistic context. After the synthesis, the output of the method is the final folksonomy with linguistic context obtained from dialogues, which in this case was evolved.

Experiments

We designed an experiment to verify if the folksonomies with linguistic context, learned by the C-FolksDialogue, improve the process of interpreting dialogue utterances, when compared to the interpretation performed by dialogue folksonomies without taking advantage of such contextual information. Note that, in this research the interpretation of utterances means to verify whether or not they belong to the domain that a learned folksonomy represents.

To perfom our experiments we used Java and the Stanford CoreNLP tool (Manning et al. 2014) for handling all the Natural Language Processing tasks in our method. First, we learned a folksonomy using a dialogue corpus in the domain of Movie-Ticket Booking ((Li et al. 2016) ; (Li et al. 2018)). For the experiment, we extracted from this corpus 800 dialogues composed of 7783 utterances, created by one interlocutor of the type attendant, together with 800 different interlocutors of the type users. We make the assumption that each of the dialogues is composed by a different user, since users are not identified. On the other hand, we assume that all the dialogues have the same interlocutor of the type attendant identified in the corpus as "agent."

For learning the folksonomy we adopted a holdout approach (Tan, Steinbach, and Kumar 2005), i.e., 2/3 for training and 1/3 for the test of the model. In the 1/3 of the dialogues used in the process of interpreting utterances, only the utterances from the interlocutors of the type users are used. The reason for do not use the attendants utterances in these step, is due the assumption that they have the full knowledge of the domain. Moreover, to test better the robustness of our approach, we enhanced the test set by adding more 20% (of its size) of utterances showing ambiguity with the meaning of the folksonomy tags. For selecting the other meanings of the tags we used the WordNet (Fellbaum 1998).

For learning the folksonomy, we applied the 532 dialogues as input of the C-FolksDialogue method, obtaining 2260 utterances of the attendants (49.99% of all utterances). From these, the method has extracted 506 unique nouns, which have formed the list Ls. Through a cutoff frequency of 5.79, from all nouns of Ls 348 (68.77%) became tags of the folksonomy. Then, through a cutoff frequency of 2.07, from the total of the utterances of the attendants, 416 (18.41%) became resources of the folksonomy. Concerning the relationships between the tags, based on a cutoff frequency of 2.38 the method has created 268 relationships between the tags. Next, we measured the accuracy of the folksonomy in interpreting utterances. From the 3262 utterances that are part of the 268 dialogues intended to this experiment, we have extracted all the 1631 utterances from the interlocutors of the type users, and then added 326 utterances (20%) with ambiguity, totalizing 1957 utterances.

The first approach (without linguistic context) for interpreting utterances was similar we did in (Wanderley et al. 2015). First, we extracted the nouns of an utterance being interpreted. Then, we verified how many of these nouns are tags of the folksonomy, by using the Ratio of Inclusion (Equation 1). The utterances with their Ratio of Inclusion, greater than or equal to the cutoff frequency applied for obtaining the folksonomy resources, were considered as belonging to the domain. After calculating the Ratio of Inclusion for each utterance, we compared these results with labels assigned by a domain expert. This comparison was done by calculating the accuracy through the Equation 2, where $nCorrect$ is the number of utterances that had the result equals to the label assigned by the expert, and n is the total number of utterances that are being tested.

$$Accuracy = \frac{nCorrect}{n} \times 100 \tag{2}$$

In the second approach (using linguistic context), first, for each test utterance we extracted its nouns and then verified if some of them was a tag. If there were no tags, then the utterance was considered outside the domain. Otherwise, for each noun that was a tag we extracted all of its dependency relations in the utterance, comparing them to the corresponding linguistic context in the folksonomy. If there was a match between some of the dependencies and the linguistic context, then the utterance was considered in the domain. Otherwise, the approach used the relations between the resources to take the linguistic context of the neighbors of the matched tags. If the linguistic context of the neighbors matched with some of the dependencies of the utterance, then it was considered in the domain, otherwise it was pointed outside. Finally, analogous to the first approach we compared the results with domain expert labels, and then computed the accuracy (Equation 2). Table 1 shows the accuracy of both approaches.

Comparing the results, one of the main conclusions we can draw is that the linguistic context resulting naturally from what the interlocutors tried to express in their utterances, supports the disambiguation of the folksonomy tags.

Table 1: The accuracy of the folksonomy for interpreting utterances.

	Without Linguistic Context	Using Linguistic Context
Accuracy	72.77%	76.41%

This context represents explicitly a meaningful (grammatical, not random) context for supporting the sense of the tags. Going deeper, utterances showing ambiguity with the folksonomy tags were considered outside the domain. For instance, for the tag "preference" the utterance "My preference is for books" was outside the domain, because it did not have context elements of the tag such as "direct object" with the verb "to have," or "nominal modifier" with the noun "rating." On the other hand, the same utterance was considered in the domain with the first approach (without context), because it had two nouns (one tag), and thus in the range of Ration of Inclusion. Moreover, the linguistic context was responsible for ensuring that there was a truly relation between a tag and a noun, i.e., not being correlated randomly, when performing the interpretation. For example, in the utterance "I have a preference on genre or rating" the tag "preference" was related with "genre" through a "nominal modifier."

Conclusion and Future Work

In this paper, we presented the C-FolksDialogue method which uses linguistic context for learning folksonomies from task-oriented dialogues. The folksonomies learned with our method can be useful to reduce ambiguity when interpreting utterances in a dialogue system.

The main contribution of our approach was to introduce the concept of linguistic context in the folksonomies learned from task-oriented dialogues. Moreover, another novelty of our method was to allow dialogue folksonomies to evolve, thus avoiding learning them every time from scratch.

We tested the linguistic context of our folksonomies in the task of interpreting utterances. Our results show that the C-FolksDialogue folksonomies improve the interpretation of utterances, indicating whether they belong or not to the domains that the folksonomies represent. Although folksonomies are simple forms of knowledge representation (compared to ontologies), the linguistic context added by our method supported them for interpreting utterances, increasing reasonably the accuracy whether compared when not using such a contextual information.

Currently, we are testing our method with other dialogue corpora. In near future, we plan to use the new linguistic contextual information enriching the folksonomies to support us to detect trends in the dialogues used to learn them.

Acknowledgments

Gregory Moro Puppi Wanderley would like to thank CNPq-Brazil (grant 153497/2018-1) for its support in this research. Moreover, the authors are very thankful to the anonymous reviewers of this paper for their kind comments and contributions.

References

Artstein, R.; Traum, D. R.; Boberg, J.; Gainer, A.; Gratch, J.; Johnson, E.; Leuski, A.; and Nakano, M. 2017. Listen to my body: Does making friends help influence people? In *FLAIRS Conference*, 430–435.

Burtsev, M.; Seliverstov, A.; Airapetyan, R.; Arkhipov, M.; Baymurzina, D.; Bushkov, N.; Gureenkova, O.; Khakhulin, T.; Kuratov, Y.; Kuznetsov, D.; et al. 2018. Deeppavlov: Open-source library for dialogue systems. *Proceedings of ACL 2018, System Demonstrations* 122–127.

Echarte, F.; Astrain, J. J.; Córdoba, A.; and Villadangos, J. E. 2007. Ontology of folksonomy: A new modelling method. *SAAKM* 289:36.

Fellbaum, C. 1998. *WordNet: An electronic lexical database*. Cambridge, MA: MIT Press.

Jurafsky, D., and Martin, J. H. 2014. *Speech and language processing*, volume 3. Pearson London.

Li, X.; Lipton, Z. C.; Dhingra, B.; Li, L.; Gao, J.; and Chen, Y.-N. 2016. A user simulator for task-completion dialogues. *arXiv preprint arXiv:1612.05688*.

Li, X.; Panda, S.; Liu, J.; and Gao, J. 2018. Microsoft dialogue challenge: Building end-to-end task-completion dialogue systems. *arXiv preprint arXiv:1807.11125*.

Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J.; Bethard, S. J.; and McClosky, D. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, 55–60.

Peters, I. 2009. *Folksonomies. Indexing and retrieval in Web 2.0*. Walter de Gruyter.

Plangprasopchok, A., and Lerman, K. 2009. Constructing folksonomies from user-specified relations on flickr. In *Proceedings of the 18th international conference on World wide web*, 781–790. ACM.

Schmitz, C.; Hotho, A.; Jäschke, R.; and Stumme, G. 2006. Mining association rules in folksonomies. In *Data science and classification*. Springer. 261–270.

Sparck Jones, K. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation* 28(1):11–21.

Strohmaier, M.; Helic, D.; Benz, D.; Körner, C.; and Kern, R. 2012. Evaluation of folksonomy induction algorithms. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3(4):74.

Tan, P.-N.; Steinbach, M.; and Kumar, V. 2005. *Introduction to data mining*. Addison-Wesley. ISBN-10: 0321321367, pp.186-187.

Wanderley, G. M. P.; Tacla, C. A.; Barthès, J.-P. A.; and Paraiso, E. C. 2015. Knowledge discovery in task-oriented dialogue. *Expert Systems with Applications* 42(20):6807–6818.

Xu, Z.; Tifrea-Marciuska, O.; Lukasiewicz, T.; Martinez, M. V.; Simari, G. I.; and Chen, C. 2018. Lightweight tag-aware personalized recommendation on the social web using ontological similarity. *IEEE Access* 6:35590–35610.