One of the great challenges of robotics is effective human-robot communication. It is quite challenging for autonomous robots to understand human natural language statements, as these must be mapped from a large highly open space of semantic constructs to precise grounded representations that robots can interpret. Probabilistic models and inference algorithms for natural language processing (NLP) have proved to be especially useful and attractive, due to their high degree of modularity, flexibility, computational efficiency, and ability to naturally tie into probabilistic reasoning algorithms for robotic decision-making [1–8]. While state-of-the-art developments in probabilistic methods have largely focused on natural language command and control, there is also great interest in the ability of humans to quickly and efficiently communicate useful sensory information, i.e. to act as ‘human sensors’ that support probabilistic robotic perception, learning and state estimation tasks [9–15].

The problems of exploiting human natural language for either sensing/estimation or planning/control have much in common, but also lead to different technical challenges for modeling and reasoning. Specifically: whereas planning/control ultimately requires identification of (constraints on) a single most likely set of executable actions to minimize some cost function, sensing/estimation tasks generally require updating complete beliefs over all possible latent state variable hypotheses. The former case can be solved with state-of-the-art point-based optimization strategies, but the latter requires efficient pdf representation and update solutions to tackle the full Bayesian inference problem.

This work describes progress toward a probabilistic framework for translating free-form natural language observations provided by human sensors into data likelihood functions for efficient online Bayesian human-robot sensor fusion. Similar to, and in parallel with, frameworks such as DCG and HDCG for planning/control [1–3], our goal is to develop a generalizable ‘plug-and-play’ interface between existing robotic Bayesian state estimators and off-the-shelf natural language processors. A key requirement to this end is to capture uncertainties related to both semantic ambiguities and lexical parsing of natural language. In contrast to state-of-the-art planning approaches, we propose a ‘loosely-coupled’ inference approach that conditionally separates the problems of inferring lexical/translation uncertainties and inferring semantic meaning of natural language for state estimation. A key challenge to this end is to derive statistically consistent/conservative estimates of lexical parser uncertainties from existing off-the-shelf probabilistic NLP components (e.g. syntactic parsers, sense-matchers), so that these can be combined with existing multiple hypothesis semantic data fusion techniques. We describe a motivating application for our work, and provide a general problem statement for mapping free-form natural language chat data provided by humans into large dictionaries of recognizable semantic soft data. We propose a generic processing pipeline for estimating required lexical parser uncertainties, and present a preliminary approach and results for estimating lexical uncertainties due to sense-matching ambiguities using pre-trained/off-the-shelf word2vec models [16].

II. PROBLEM FORMULATION, AND RELATED WORK
A. Motivation: ‘Cops and Robots’ Dynamic Target Search
As a grounding application, we have developed an indoor autonomous target search testbed for the ‘Cops and Robots’ scenario, in which one ‘security cop’ robot (Deckard) searches for multiple ‘robber’ robots, with the assistance of a human ‘deputy’. The indoor search environment consists of one mobile cop robot, 3 mobile robber robots (Roy, Pris, and Zhora), and semantic reference features on a known map, e.g. dividing rooms, areas, furnishings, etc. Each robot uses an iRobot Create base for mobility, a Kinect camera for onboard sensing, and an Odroid U3 on-board computer running ROS. The human deputy is located remotely and uses an interface shown in Figure 1 to communicate with the cop robot. All robots and map objects are localized by a Vicon system. The human deputy can see the cop robot’s camera feed, or select a view from one of three security cameras in the environment.

In the initial interface, the human selects from a fixed list of possible semantic observations (lower-left of Fig. 1) to send a structured message to the cop, of the form

\[
\langle \text{TargetID} \rangle \langle \text{is/is not} \rangle \langle \text{Preposition} \rangle \langle \text{Landmark} \rangle , \quad (1)
\]
where the field elements are taken from a known dictionary. Alternatively, structured semantic questions generated by the cop can be answered by the human, where the top 5 questions formed from the fixed list elements are ranked according to maximum expected information gain (posterior entropy decrease) for a given target, e.g. ‘Is Roy in the dining room?’

The interface shows each target’s evolving location pdf as a heat map over the environment. The cop fuses onboard visual information (limited range target detection/non-detection events for search and range/bearing data for tracking) with semantic soft data provided by a human. For now, as in [10, 17], the cop separates the motion planning and data fusion problems; the human has no direct control over the cop’s movement, but can influence the cop’s beliefs in where targets are (and are not) located. As such, the cop must autonomously decide how best to utilize the information it receives. In doing so, it must also cope with the fact that its probabilistic beliefs over dynamic robotic locations will generally be described by non-convex/multimodal pdfs, due to complex target motion and sensing uncertainties.

B. Formal Bayesian Human-Robot Data Fusion

Previous work in [10, 18] established a rigorous, formal method to identify likelihood functions for semantic soft data provided by human sensors and (recursively) fuse these with conventional robotic sensor data. We first provide a brief summary of this approach using general dynamic state space models. For a discrete time index $k \in \mathbb{Z}^+$, let $X_k \in \mathbb{R}^n$ be the uncertain continuous target state vector (e.g., position, heading, velocity). Given a prior belief $p(X_{k-1})$, potentially conditioned on ‘hard’ robot sensor data $Z_k$ and known target dynamics $p(X_k|X_{k-1})$, soft data $D_k$ can be fused to obtain the posterior target state pdf via the Bayes filter:

$$p(X_k) = \int p(X_k|X_{k-1})p(X_{k-1})dX_{k-1}$$

where $p(X_{k-1}) \propto P(D_k|X_k)p(X_k)$.

The soft data likelihood $P(D_k|X_k)$ represents a hybrid (i.e. continuous-to-discrete) mapping between categorical human sensor statements and intrinsically uncertain semantic classification of the true state. We model $P(D_k|X_k)$ through generalized softmax functions [18–20],

$$P(D_k = l|X_k) = \exp (w_l^T X_k + b_l)/\sum_{j=1}^m \exp (w_j^T X_k + b_j)$$

where $l$ selects one of $m$ possible categorical semantic labels, each parametrized by weights $w_l$ and biases $b_l$ which can be learned from experimental data. Fig. 2 shows an example softmax model with 17 spatial relation classes. As discussed in [18, 19], generalized softmax models provide a rich convenient polytopic geometry for easily embedding a priori contextual information (e.g. object/environment geometries, dynamic constraints), leading to significant parameter sparsification and greatly mitigating expensive non-convex optimization procedures for model identification. Furthermore, the product of generalized softmax likelihoods from multiple independent observations can be quickly compressed into a single likelihood for efficient online ‘batch’ data fusion. When $p(X_k)$ is a Gaussian mixture (GM),

$$p(X_k) = \sum_{p=1}^{m_p} w_p N(x_k|\mu_p, \Sigma_p)$$

(where $w_p$, $\mu_p \in \mathbb{R}^n$, and $\Sigma_p$ are the weights, mean vector and covariance matrix for mixand $p$), the posterior (3) is well approximated by another GM via combined variational inference and importance sampling techniques, enabling fast recursive state estimation with multimodal pdfs [10]. GM compression methods [21] can be used to efficiently manage mixture growth from multimodal transition pdfs $p(X_k|X_{k-1})$ (e.g. unobserved switching dynamics) or semantic likelihoods $p(D_k|X_k)$ (e.g. non-convex range-only semantic data). As such, the GMs enable (parallelizable) maintenance of complex belief hypotheses over the whole state space for approximate Bayesian inference. Fig. 3 shows the dynamic Bayesian network (DBN) model for the soft data fusion process for an arbitrary dictionary of $m$ semantic observations at any given time step. Each $D_k^s$ term has its own generalized softmax likelihood to describe semantic state space uncertainty for associated category labels (e.g. $D_k^s$’s label set may describe robber bearing relative to the cop, $D_k^r$’s label set may describe robber velocity, etc.), and shaded nodes indicate actual observation data provided by a human.

C. Structured Interfaces for Reporting Semantic Soft Data

A human could report soft data by constructing structured observations $l$ from a list of available statements, as shown lower-left in Fig. 1. This bypasses the difficult problem of parsing natural language inputs, but leads to several major limitations. Firstly, it restricts the human to a rigid pre-defined dictionary and message structure, which may not provide an intuitive or sufficiently rich set of semantics to convey desired information. Secondly, it is very inconvenient to select items one-by-one from a list for structured messaging; especially as the dictionary size grows, this quickly becomes infeasible and time-consuming enough to render data irrelevant in dynamic settings. Furthermore, this approach does not scale well with environment/problem complexity and lexical richness for soft data reporting. In particular, it is often desirable to provide observations that activate multiple $D_k^s$ terms (bottom of Fig. 3), e.g. ‘Roy is by the table and heading slowly to the kitchen’ simultaneously provides location, orientation and velocity information.
D. Proposed Natural Language Chat Interface

To address these limitations in the Cops and Robots application, we are developing an unstructured natural language chat interface to support highly flexible ‘free-form’ soft data reporting. The chat interface should ideally support a wide range of semantics and enable fast transmission of rich soft data reports. However, the problem of deriving meaningful and contextually relevant dynamic state information from a free-form chat observation $O_k$ is highly non-trivial. Unlike with structured messages, it is infeasible to explicitly construct likelihood functions $p(O_k|X_k)$ in advance to find the Bayes posterior $p(X_k|O_k)$ directly. Furthermore, many different chat messages can convey exactly the same or slightly different varieties of information, leading to additional uncertainties in lexical meaning (i.e. possible translation errors) in addition to intrinsic semantic (state spatial) uncertainty. For example, the phrases ‘Blue robot moving past the books’, ‘Roy next to the bookcase going to kitchen’, and ‘Robber nearby shelf heading left’ all overlap in the sense that they could all essentially refer to a structured set of atomic phrases such as ‘Roy is near the bookcase; and Roy is moving toward the kitchen’.

Since similar issues are also encountered in natural language interpretation for planning/control, modeling approaches developed for that domain could be adopted here. In approaches such as G$^3$ [5] and more recently DCG and HDCG [1–3], MAP-based inference is performed over a large set of random variables defined by factor graphs to determine a single most likely set of latent semantic groundings and desired actions/action constraints from natural language command inputs. These models consider dense overlapping sets of hierarchical and undirected conditional dependencies that comprehensively capture the complex relationships between different possible linguistic meanings and contexts within a modular joint probability distribution. Due to the complexity of the joint pdf, it is non-trivial to represent the full posterior distribution over unknown variables and so point-based solutions are sought instead. Inference operations over continuous random variables are also typically approximated via discretization, which is suitable for online planning scenarios for low state dimensionality. These features impose significant operational limitations from the Bayesian human-robot sensor fusion perspective, as autonomous robots will generally want to reason over and track all possible likely state hypotheses for complex applications, especially if these are significantly multimodal and have many dynamic states as in the Cops and Robots scenario.

Furthermore, state-of-the-art NLP-to-planning models essentially rely on ‘tightly coupled’ inference, i.e. in analogy to tightly-coupled GPS-based vehicle navigation systems which comprehensively account for and reason jointly over co-dependent uncertainties in vehicle, receiver, satellite and atmospheric propagation error parameters and states [22]. Such reasoning is arguably quite essential for high precision and ‘single-shot’ applications, where information-gathering opportunities are limited. This motivates consideration of simpler ‘loosely coupled’ inference architectures for other cases where coarser-grained and/or multi-shot reasoning is sufficient. By ‘loosely coupled’, we mean to separately account for lexical/translation uncertainties and semantic/meaning uncertainties in a statistically consistent way that avoids joint reasoning over a large set of latent variables. Ideally, such a reasoning framework could be designed and implemented with off-the-shelf NLP components (e.g. probabilistic syntactic parsers and sense-matchers) to provide (hopefully conservative) ‘lumped’ approximations of lexical/translation uncertainties, while semantic/meaning uncertainties for a given lexical parsing are handled by a state fusion engine.

To build on the Bayesian GM fusion framework for state space models, our approach is to translate a given $O_k$ into a reasonable ‘on the fly’ estimate of the likelihood $p(O_k|X_k)$ via a very large (possibly expandable) dictionary of latent $D_k$ semantic observations, which have known generalized soft-max likelihoods $p(D_k|X_k)$. In particular, given the expansion (where $O_k$ is conditionally independent of $X_k$ given $D_k$)

$$p(X_k|O_k) \propto p(X_k) \sum_{i=1}^{m} \sum_{j=1}^{m_i} p(O_k|D_k^i = j)p(D_k^i = j|X_k),$$

we can generally approximate $p(O_k|D_k^i)$ as the second summation term on the RHS, where $p(O_k|D_k^i = j)$ accounts for the lexical uncertainty and $p(D_k^i = j|X_k)$ accounts for the semantic uncertainty. Since $O_k$ may also point to multiple soft observations (i.e. data about position and heading data for a target in the same sentence), the $D_k^i$ likelihoods on the RHS generally could correspond to unique products of independent dictionary terms, e.g. as in the multiple instantiations for the bottom row of Fig. 3. The general problem, then, is to identify how $D_k^i$ likelihoods (or sets of soft observations) should be ‘activated’ for a given $O_k$ input by identifying the scalars $p(O_k|D_k^i = j)$. Since $p(X_k)p(D_k^i|X_k)$ can generally be approximated as a GM, it follows then that the LHS $p(x_k|O_k)$ generally corresponds to a ‘mixture of mixtures’ (that is, as long as $O_k$ could ‘truly’ match to only one set of soft observations, so that $p(x_k|O_k)$ sums to one).

III. METHOD AND PRELIMINARY RESULTS

Fig. 4 shows a generic serial NLP pipeline for translating $O_k$ into $D_k$, which can be wrapped around the semantic data fusion engine to provide estimates for lexical uncertainty terms $p(O_k|D_k^i)$ given a large fixed semantic dictionary defining possible $D_k$ soft data primitives. We assume that the soft data primitives can be predefined offline for a known environment with a fixed number of possible groundings and a finite number of feasible semantic relations defined in the state space $X_k$. However, this does not restrict the ability to augment the dictionary or modify semantic likelihoods $p(D_k^i|X_k)$ online, e.g. to describe spatial relations for moving landmarks (e.g. the cop robot). Each component layer of the pipeline represents a key step for translating free-form chat inputs into recognizable semantic information, and thus captures a potential source of lexical parsing errors, which percolate down to the next layer. A key point is that each component layer can, in principle, be instantiated by any number of existing off-the-shelf NLP tools. Internal components include: a tokenizer, which segments a text stream into multi-word tokens; a tagger, which tags each token with a semantic label; a templater which fits label-token pairs into pre-defined sensor statement templates; and a sense-matcher, which maps natural language tokens to the list of pre-defined tokens that populate appropriate positions in sensor statement templates. Fig. 7 shows the general sensor statement template, which uses token labels derived from pre-defined Target Description Clauses (TDCs), which are conceptually similar to Spatial Description Clauses.
A. Sense-matching with label variations to the closest matching dictionary terms.

for D the semantic labels that may be outside of the fixed dictionary of speech for a valid soft data statement, but have variations in processed to the point where they should match expected parts the following, we consider unstructured chat inputs that are tion, tagging and templating has been performed. Hence, in
of the pipeline, under the assumption that perfect tokeniza-
uncertainty estimates from the final sense-matching portion
∫

Fig. 4: A proposed NLP pipeline (left) with an example (right) for loosely coupled soft data fusion.

(SDCs) [6]. Different TDC types consist of different compo-
teints that form sub-trees of Fig. 7; for instance, an action
tDC, composed of a certainty (how sure the human is of the observation), a target (none, one or multiple entities), a positivity (e.g. ‘is’, ‘is not’), and action (description of the target motion).

In theory, provided that the possible outputs of each component layer can be assigned a conditional probability of being correct given the outputs of the previous component layer, then \( p(O_k|D^k) \propto p(D^k_1|O_k)/p(D^k_1) \) can be obtained (up to constant \( p(O_k) \)) by estimating \( p(D^k_1|O_k) \) as the product of the component layer conditional probabilities (i.e. for parsing events leading up to some \( D^k_1 \)), and finding \( p(D^k_1) = \int p(X_k)p(D^k_1|X_k)dx_k \) from the corresponding fusion result. Due to limited space, we focus attention here on obtaining uncertainty estimates from the final sense-matching portion of the pipeline, under the assumption that perfect tokenization, tagging and templating has been performed. Hence, in the following, we consider unstructured chat inputs that are processed to the point where they should match expected parts of speech for a valid soft data statement, but have variations in the semantic labels that may be outside of the fixed dictionary for \( D^k_1 \) terms. Hence, the problem is to map these semantic label variations to the closest matching dictionary terms.

A. Sense-matching with word2vec

The key issue in mapping between unstructured language and a set of structured semantic templates is the derivation of a relationship between the estimated 13 million tokens in the English language and the comparatively minuscule number of tokens in a set of predefined template statements. Recent efforts, particularly by Mikolov et al. [16], introduced a negative sampling-based approach to efficiently learn multi-dimensional vector representations of all tokens in a vocabulary. Using this, word2vec tool, we can develop a mapping between a set of tokens contained within an unstructured utterance and a set of tokens contained within a structured semantic template.

Each structured statement \( D_k \) can be decomposed into tokens \( d_{k,1}, d_{k,2}, \ldots d_{k,N_k} \). If we are given the tokenization \( T_k \) of an unstructured statement, decomposed into tokens \( t_{k,1}, t_{k,2}, \ldots t_{k,N_T} \), we can form a mapping between the tokens of two statements. This relationship is captured graphically in

Fig. 6: Token-matching problem reformulated as a Hidden Markov Model.

One of the key features of this assumption is that we now impose a one-to-one pairing between tokens of structured and unstructured phrases, and thus that both phrases are tokenized to the same length. If the prior elements of the pipeline are able to provide this tokenization, we only require transition probabilities \( P(d_{k+i+1} \mid d_{k,i}) \) and observation probabilities \( P(t_{k,i} \mid d_{k,i}) \) to specify the conditional probability of each structured statement \( P(D_k \mid T_k) \).

We can use known structure to determine the transition probabilities. Figure 7 illustrates a possible structure for a human sensor statement. Each node is a template, containing one or more possible tokens; for instance, the Spatial Relation: Object node may contain, ‘front’, ‘left’, ‘back’, ‘right’ and ‘near’, whereas the Spatial Relation: Area node may contain ‘inside’, ‘near’ and ‘outside’. If each template node is expanded into a set of token nodes, then each path through the

Fig. 7: Example human sensor statement template structure; each node decomposes into a set of possible tokens to generate each template structured statement.
tree from the root to a leaf node generates one statement. Thus, transition probabilities between \( d_k \) and \( d_{k+1} \) can be derived by parent-child relationships:

\[
P(d_{k,i+1} \mid d_{k,i}) = \begin{cases} \frac{1}{k} & \text{if PARENT}_{d_{k,i+1}} = d_{k,i} \\ 0 & \text{otherwise} \end{cases}
\]

(5)

where \( \nu \) is the number of tokens captured by the template node that contained \( d_{k+1} \). This assumes all possible paths are equally likely within the constraints specified by the template.

Calculation of the observation probabilities asks the question, ‘What is the probability of observing the token \( t_{k,i} \), given that I have selected the template token \( d_{k,i} \)?’ This question relates to the core issue of lexical uncertainty in our problem: the probability of observing, for instance, ‘near’ given that the template word is ‘near’ is non-unity; furthermore, since \( \sum_{t_{k,i}} P(t_{k,i} \mid d_{k,i}) = 1 \) and \( t_{k,i} \) spans the English language, we notice little discrimination between two words from the conditional probability.

At its core, the Skip-gram based \textit{word2vec} uses a softmax function (c.f. [16] eq. 2) to relate the probability of one word, \( w_i \), represented by its input and output vectors \( v_i \) and \( v'_i \), given another word, \( w_j \), represented by its input and output vectors \( v_j \) and \( v'_j \):

\[
p(w_i \mid w_j) = \frac{\exp(v'_i^T v_j)}{\sum_{w=1}^W \exp(v'_w^T v_j)}
\]

(6)

Since \( W \) is the number of words in the vocabulary (likely in the millions for unstructured language), \( p(w_i \mid w_j) \) will be both small and nearly flat over the vocabulary. A common measure of word similarity, in place of conditional probability, is to instead take the cosine similarity measure between output word vectors. This similarity measure need not sum to 1 for all words and is exactly 1 for identical words. A first step approach would be to replace the token observation probability \( P(t_{k,i} \mid d_{k,i}) \) with a token observation similarity score \( s(t_{k,i}, d_{k,i}) \), weight all possible statements by their joint scores:

\[
s(D_k, T_k) = P(d_1)s(t_1, d_1)P(d_2 \mid d_1)s(t_2, d_2) \times \ldots \times s(d_{N_T}, d_{N_T-1})P(t_{N_T} \mid d_{N_T})
\]

(7)

We can now use \( \arg \max_{D_k} s(D_k, T_k) \) to select template sensor statements that are most similar to the input tokenization. A proof-of-concept example, seen in Fig. 8, shows the results of generating \( s(D_k, T_k) \) for some test phrases. In this example, 2682 possible template statements are considered, which covers 79.81% of the 208 input sentences collected by our pilot experiment. Four input phrases are shown and assumed to have been tokenized correctly by previous pipeline elements to demonstrate the sense matching portion of the pipeline independently. From left-to-right, the four columns demonstrate the effects of increasing dissimilarity with template statements: the first input sentence is exactly a sensor statement template; the second input sentences replaces a spatial relation template token, ‘near’, with a non-template token, ‘next to’; the third input sentence replaces the grounding and changes the positivity; and the fourth is an imprecise reformulation of the first sentence. These results are promising, as the top-scoring statements are all qualitatively similar to the original phrase.

IV. CONCLUSION AND ONGOING WORK

This work describes initial progress toward a probabilistic framework for translating free-form natural language observations provided by human sensors into recognizable data likelihood functions for efficient online Bayesian human-robot sensor fusion. We proposed a ‘loosely-coupled’ inference approach that conditionally separates the problems of inferring lexical/translation uncertainties and inferring semantic meaning of natural language for state estimation. A key challenge to this end is to derive statistically consistent/conservative estimates of lexical parser uncertainties from existing off-the-shelf probabilistic NLP components (e.g. syntactic parsers, sense-matching), so that these can be combined with existing multiple hypothesis semantic data fusion techniques. We described a motivating application for our work and a general problem statement for mapping free-form natural language chat data provided by humans into large dictionaries of recognizable semantic soft data. We presented a generic processing pipeline for estimating required lexical parser uncertainties, and presented a preliminary approach for estimating lexical uncertainties due to sense-matching ambiguities using \textit{word2vec} models.

Ongoing work includes further analysis and exploration of \textit{word2vec} and newer related tools, e.g. \textit{sense2vec}, to generate more context-sensitive lexical parsing probabilities. Although preliminary results are not provided here, we are also investigating how a number of other off-the-shelf NL parsing tools and probabilistic techniques can be used to generate the remainder of lexical parsing uncertainties in the proposed pipeline, e.g. including SpaCy for syntactic parsing (https://spacy.io/docs) and conditional random fields for label tagging [23]. Important future directions include exploring connections to factor graph models such as DCG and HDCG to derive more rigorous estimates for lexical uncertainties, e.g. using staged conditional inference approximations in combination with GM-based semantic fusion updates. Finally, we will explore the rich space of techniques developed within the sensor fusion community for fusing data under unknown information correlations, which are particularly important for coping with unknown dependencies or inconsistencies between off-the-shelf parser components [24]. These also include multi-hypothesis data association methods, which could permit delayed data fusion through sequential scoring of multiple parallel lexical parsing histories over time.

REFERENCES


