

# Using Chapel for Natural Language Processing and Interactions

Author: Brian Guarraci (CTO, Cricket Health)

Presenter: Brian Guarraci

Natural language interfaces, such as Siri, are quickly making advances into consumer products and applications. One common application of natural language interface is the chat interface to a conversation engine, or chatbot. Responding to user input can be treated as a kind of specialized search, so in order to make a conversation engine useful it must efficiently access substantial language and knowledge bases. Using lessons learned from Chearch, a Chapel-based text search engine, and Chapel's inherent capabilities, I present two methods of preparing data for a conversation engine. The two methods of interest are word embeddings and graph-based representations.

A popular Natural Language Processing (NLP) model is word embeddings, in which text corpus is transformed into vector representation of words. Computing word embeddings is a highly computational task and benefits from Chapel's capabilities by enabling both larger output vectors and larger raw datasets. I will present a Chapel port of Google's popular word2vec (1) application. The existing word2vec application supports multiple threads on a single machine, while Chapel enables the algorithm to scale to a cluster.

Graph-based representations of text are another powerful way to capture relationships between entities. Chapel's distributed computing abilities make it straightforward to both create large graphs as well as apply algorithms that operate at scale. I will present experiments with both how to build graph-based representations of text using Chapel and apply algorithms to create custom datasets for a conversation engine.

Although it would be ideal to write a conversation engine in Chapel, for this presentation I will present how prepared data can be used with a successful chatbot framework called ChatScript (2). By leveraging Chapel to process large quantities of raw data and producing domain specific datasets, a chatbot can be customized for particular applications.

References:

1. Word2vec: <https://code.google.com/archive/p/word2vec/>
2. ChatScript: <https://sourceforge.net/projects/chatscript/>