**Google FRA Proposal**

**Snippet learning through word vectors & context descriptors**

**Abstract:** While over the past decade 'Big Data' has become a very important topic within the Machine Learning community, we should not overlook the fact that 'Small Data' is a much harder problem to solve. Consider a small amount of text (snippet) consisting of only two sentence parts: ‘I am in an airplane - will be back in two weeks’: Depending on whoever sends this message it can have very diverse meanings ranging from 'I am on vacation' to 'I am on a conference' or even 'I want a divorce, honey'. The meaning of the snippet thus lies less in the words or phrases themselves, but largely within the context applied to them. Since snippets do not offer the information content required for frequency-based learning approaches, we will develop a two-part learning methodology: First, snippet descriptors will be constructed out of word vectors gained from pre-trained vector collections (glove, word2vec). Second, we strive to construct expressive but computationally efficient human entity descriptors (HEDs) to combine with the previously obtained snippet descriptors, thus enabling context-aware learning even on small, mobile devices or within web browsers. Finally, we are going to apply this methodology to a body of text consisting of individual units of relatively short length, like Tweets or messages in a chat application. As a medium-term business case, we envision goal extraction from free text as a basis for personalized strategy recommendation, a task still reserved for human consultants up to now.

**Introduction:** Natural Language processing has been around for at least the past 66 years (counting from Alan Turing’s paper ‘Computing Machinery and Intelligence’ in 1950), and has achieved ever better results in recent years, especially with the advent of generative models like Latent Dirichlet Allocation (LDA) which see topics as kind of ‘fountains’ producing word distributions. A recent advancement in NLP was achieved via the introduction of word vectors [1] [2], an approach which not only uses a Vector Space Model (VSM) for word representation, but also encodes word vectors (with axes representing concepts or a mix of concepts) numerically so that computations can be conducted on them. The canonical example for this is the ‘king’ vector from which the ‘man’ vector can be subtracted and the ‘woman’ vector added, producing a result very close to the ‘queen’ vector.

**Problem description:** While frequency-based methods of NLP (TF-IDF, Bag of Words) work very well on large corpora of text, they fail to extract meaning from small data like text snippets, since there is not enough meaning contained in the words themselves. Let’s consider the example given above: a text snippet saying ‘I am in an airplane – will be back in two weeks’: If the person sending this text is a young woman with an extensive circle of friends all over the world, it is likely that this person is going on a vacation – related concepts coming to mind would be ‘relaxation’, ‘beach’, ‘sailing trip’ etc. If on the other hand the text originates from a professor who is known for his aversion against leisure and has extensive relations with Columbia University, the meaning of the text will radically shift. Therefore, the simple wording of a text snippet acts only as a trigger to a much more powerful datastructure in the mind of the reader, which we intend to model and demonstrate its use in several interesting experiments as well as a full-blown business case.

**Methodology:** Our idea is to model this background knowledge in the form of a context-graph [3] [4], which is part of a larger concept known as ‘ambient intelligence’. A context graph is in itself a recommender system for finding new materials based on the existing context of a user. In our case this graph will contain several node types such as people, locations, hobbies etc. Furthermore, we will combine those graphs with a word-vector based approach applied to small snippets of text. In this approach, word vector collections will first be trained on standard corpora of text, yielding a concept space in the few-hundred-dimensional range, reduced from the original vocabulary space of several to hundreds-of-thousands of dimensions (depending on the input text). Subsequently for each word in the snippet it’s respective vector will be applied, leading to an overall interpretation of the text under question. While the exact method of combining these components is still not perfectly clear, a likely approach would be the inclusion of additional word-vectors of related concepts as queried from our context graphs, as well as a weighting of word-vectors according to contextual probabilities.

**Scenarios / Variations:**

1. Different snippet descriptors: As word vectors only represent individual words through their context, we need to find a way to combine those vectors in order to represent a whole text segment. A method as simple as vector addition might already yield acceptable results, but more realistic scenarios would include the construction of a convex hull in concept space or an entirely different data structure such as graphs, trees or state models (with states representing concepts).
2. Word vectors trained on a huge, general corpus like Wikipedia might yield acceptable concept vectors for every day’s text understanding – however, word-sense disambiguation remains a major problem, even more so when dealing with specialized areas. Let’s take the canonical example of a ‘king’ vector: in a traditional sense the word ‘king’ would combine concepts such as power, wealth or (sometimes) wisdom. But this is only the case when the word is seen in a political or historical context. If we shift our conversation to the field of chess, for example, the meaning of the king vector should shift as well. This can probably be achieved by training word vectors on different corpora of texts (such as history, chess or rock’n’roll texts), thereby shifting the relative concepts a term will encode depending on our requirements (see point 6 in ‘Business case’ below). It would be interesting to see how the meaning of a whole tweet or other message will be influenced if subjected to different word vector collections obtained in this way.

**Real-world experiment:** There are many situations in which ‘small’ text messages are exchanged in a way that is highly context-dependent, like tweets, sms, but even traditional email. Because our intention is to extract personal goals from text, we will focus on a form of task-centered communication like team chats since they usually revolve around solving specific, technical problems that will hint towards a team’s (or team member’s) desired achievements. The application will therefore work in several different phases:

1. Identification of people – this should be a rather easy part as it can be achieved via the usage of simple dictionaries; additionally, any ‘named-entity-recognition’ analysis should suffice for extracting organizations, locations etc.
2. We will then analyze the whole corpus of text, building context graphs for each identified person within the chat. These graphs will provide ambient intelligence for using word vectors later on.
3. Segmentation of the chat into logical units – these will probably be constituted by the individual messages, but could also be merged from several messages or segmented as a part of one single message.
4. The individual units are then cleaned according to traditional models, i.e. stop word elimination, stemming etc.
5. Applying general word vectors in combination with context graphs to messages of an individual in order to extract a person’s main goals. Once a goal (and its category – like ‘software engineering’) has been determined, we can now shift to word vectors trained on that specific category.
6. Re-analyzing an individual’s messages in accordance with his or her main goal – filtering out a series of steps that this person needs / wants to take in order to tackle the problem. This knowledge could purely emerge from the text under question or be derived from external sources of knowledge. At the end of this step we should have arrived at a series of tasks or steps with dependencies amongst another – which we define as a ‘strategy’.
7. Computing pairwise similarities amongst strategies can then help to build a task recommender capable of translating lessons learned from one user’s efforts to other users lacking further behind in their goal realization process. This can be done via task or resource recommendations, where each accepted recommendation strengthens a person’s context graph, making the system more robust with every successful interaction.

**Outcome:** The proposed, concrete outcome of our work will be a software suite capable of analyzing peoples’ goals and strategies, offering smart, personalized recommendations based on strategy similarity extracted via context graphs & word vectors applied to text snippets within a team chat setting. Moreover, we plan to publish at least 2 papers: One concerning the combination of word vectors and context graphs, and another on the problem of goal (strategy) extraction from a series of related text messages.

**Costs: 50,000 USD** for the following three items:

* **42,000** **USD** – one year’s salary for a PhD student including all related costs
* **5,000 USD** – travel costs to 2 conferences (probably ACL / EACL)
* **3,000 USD** – equipment: GPU or rented processing time for training of several word vector collections with different algorithms.

**References:**

[1] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global Vectors for Word Representation." *EMNLP*. Vol. 14. 2014.

[2] Goldberg, Yoav, and Omer Levy. "word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method." *arXiv preprint arXiv:1402.3722* (2014).

[3] Andrei Olaru, Adina Magda Florea, Amal El Fallah Seghrouchni. Graphs and Patterns for Context-Awareness. Novais, Paulo and Preuveneers, Davy and Corchado, Juan. Ambient Intelligence - Software and Applications, 2nd International Symposium on Ambient Intelligence (ISAmI 2011), University of Salamanca (Spain) 6-8th April, 2011, Apr 2011, Salamanca, Spain. Springer Berlin / Heidelberg, 92, pp.165-172, 2011.

[4] Kötters, Jens, Heinz Schmidt, and David McG Squire. "Context Graphs—Representing Formal Concepts by Connected Subgraphs." International Conference on Formal Concept Analysis. Springer Berlin Heidelberg, 2009.