Name : Mohamed Atef Abdelnaby AbdAllah Elkaffas

ALEXANDRIA

ITI_AI_45

Traditional NLP models treat words as discrete indices, ignoring similarity between them. Neural network language models (NNLMs) learn continuous “distributed” word vectors that capture both syntactic and semantic relations but are computationally expensive (dominated by the hidden-to-output matrix multiplication) . Scaling these models to billions of tokens and millions of word types was previously infeasible.

Table 6: *Comparison of models trained using the DistBelief distributed framework. Note that training of NNLM with 1000-dimensional vectors would take too long to complete.*

| Model | Vector Dimensionality | Training words | Accuracy [%] | | | Training time [days x CPU cores] |
|---|---|---|---|---|---|---|
| | | | Semantic | Syntactic | Total | |
| NNLM | 100 | 6B | 34.2 | 64.5 | 50.8 | 14 x 180 |
| CBOW | 1000 | 6B | 57.3 | 68.9 | 63.7 | 2 x 140 |
| Skip-gram | 1000 | 6B | 66.1 | 65.1 | 65.6 | 2.5 x 125 |

Table 7: *Comparison and combination of models on the Microsoft Sentence Completion Challenge.*

| Architecture | Accuracy [%] |
|---|---|
| 4-gram [32] | 39 |
| Average LSA similarity [32] | 49 |
| Log-bilinear model [24] | 54.8 |
| RNNLMs [19] | 55.4 |
| Skip-gram | 48.0 |
| Skip-gram + RNNLMs | **58.9** |

estimate since the data center machines are shared with other production tasks, and the usage can fluctuate quite a bit. Note that due to the overhead of the distributed framework, the CPU usage of the CBOW model and the Skip-gram model are much closer to each other than their single-machine implementations. The result are reported in Table 6.

### 4.5 Microsoft Research Sentence Completion Challenge

The Microsoft Sentence Completion Challenge has been recently introduced as a task for advancing language modeling and other NLP techniques [32]. This task consists of 1040 sentences, where one word is missing in each sentence and the goal is to select word that is the most coherent with the rest of the sentence, given a list of five reasonable choices. Performance of several techniques has been already reported on this set, including N-gram models, LSA-based model [32], log-bilinear model [24] and a combination of recurrent neural networks that currently holds the state of the art performance of 55.4% accuracy on this benchmark [19].

We have explored the performance of Skip-gram architecture on this task. First, we train the 640-dimensional model on 50M words provided in [32]. Then, we compute score of each sentence in the test set by using the unknown word at the input, and predict all surrounding words in a sentence.

**Efficient log-linear architectures:** Continuous Bag-of-Words (CBOW) and Skip-gram that eliminate the costly non-linear hidden layer of traditional neural language models using methods like hierarchical soft-max or negative sampling

- **Scalability** : Demonstrated training on up to 6 billion tokens with a 1 million-word vocabulary in hours to days, orders of magnitude faster than previous NNLMs.

- **State-of-the-art accuracy:** Achieved best-in-class performance on a comprehensive semantic-syntactic analogy test set, outperforming feed-forward and recurrent models.

The paper developed a word2vec tool. This tool takes a text corpus as input and produces the word vectors as output and employs two architectures ; CBOW (Continuous bag of words) and Continuous Skip gram. Because there is no non-linear transformation between projection and output, all of the "learning" happens in the embeddings themselves, and you get a **purely linear** model that can be trained on massive corpora in a fraction of the time required by traditional NNLMs.

We also notice that skip grams combined with RNNLMs provide the best accuracy.

Table 8: *Examples of the word pair relationships, using the best word vectors from Table 4 (Skip-gram model trained on 783M words with 300 dimensionality).*

| Relationship | Example 1 | Example 2 | Example 3 |
|---|---|---|---|
| France - Paris | Italy: Rome | Japan: Tokyo | Florida: Tallahassee |
| big - bigger | small: larger | cold: colder | quick: quicker |
| Miami - Florida | Baltimore: Maryland | Dallas: Texas | Kona: Hawaii |
| Einstein - scientist | Messi: midfielder | Mozart: violinist | Picasso: painter |
| Sarkozy - France | Berlusconi: Italy | Merkel: Germany | Koizumi: Japan |
| copper - Cu | zinc: Zn | gold: Au | uranium: plutonium |
| Berlusconi - Silvio | Sarkozy: Nicolas | Putin: Medvedev | Obama: Barack |
| Microsoft - Windows | Google: Android | IBM: Linux | Apple: iPhone |
| Microsoft - Ballmer | Google: Yahoo | IBM: McNealy | Apple: Jobs |
| Japan - sushi | Germany: bratwurst | France: tapas | USA: pizza |