# Competency Grading of Language Model

## Project Course in Data Science, DD2430

Supervisor: Jussi Karlgren, jussi@gavagai.se


Rithika Harish Kumar
rihk@kth.se

Rickard Kodet
kodet@kth.se

Suhas Sheshadri
suhass@kth.se


December 12, 2018


**Abstract**

Language models are used in considerable amount of applications. Word space models are one such part of language models which are used for Natural Language Processing tasks. Word Space models represent words as vectors in high dimensional spaces. Shortfall of word space models is that they do not provide an easy way to measure the level of knowledge of different topics. The aim of this project is to investigate the possibility of having a metric to automatically measure the competence of the system. It was approached by training two models; an expert and a novice, to get a better understanding of our evaluation. The most intuitive way would be to compare the frequency of words between the models. Since this does not take into consideration the context of the words, additional metrics such as similarities and rankings were also considered. To achieve this, five word categories were designed. Then, words for each categories from each topic were chosen and their performance were measured between the two models. Our result show that there is a possible way to identify a novice from an expert or at least differentiate between them. In order to arrive at a precise metric, further investigation with different data and models are required.

# 1 Introduction

In computational linguistics, we have various models trained on huge amounts data and perform project specific operations on the extracted information or features from the trained models. But, it is difficult to know whether the extracted information from the models are actually of any value. This is due to the fact that no one knows how much knowledge does the trained models have for any particular topic. In this project, we intend to find ways to measure the knowledge of a model, given some topic of interest. The ultimate goal is to give a score on the knowledge of a model for any particular topic, without knowledge of the data used for training the particular model.

The use case under consideration is e.g. a client looking to purchase a language modelling component for an information system who wishes to ensure that the model in question is trained adequately for the business sector the client is active in. The client will formulate terms which cover a number of central concepts for the field they wish to probe and the system will be tested to see if those terms are known to it.

## 1.1 Intuition

Human intuition plays a major role in understanding the context of a sentence. Assume that you are well versed with a topic and you are having a conversation. You see a new word which you have never used in this context. But, as you know about the topic and from the context of the sentence, you can have an intuition about the meaning of the word and how it relates to the topic. Machines, unfortunately, do not have the luxury of intuition. Without the human-like experience, words are just sequence of characters for them. This is the reason why we need to come up with a different way to measure knowledge of a language model.

Assume you are an expert in an area and meet someone who is also interested in the topic. How would you gauge the level of knowledge of that person? You would probably start a conversation on a higher level and slowly get deep into the topic. Without any clearly defined rules, we as humans intuitively know on how to approach this situation.

Considering the case of Badminton, here is a small snippet:
Person 1: Are you participating in the tournament next week?
Person 2: No, I just play for fun. Are you participating?
P1: Yes, I have been practicing all month.
P2: Oh, you are a professional player then?
P1: Yes, this is my fifth tournament this year.
P2: That's great. Have you won any? If yes, whats the secret behind your success?
P1: Yes, I have won three of them. Mastering the strokes, footwork and of course complete control of the shuttlecock, especially dribbling at the net does the trick for me.
P2: That is interesting. Never thought about the these while I played. Thank you.

In such a conversation, one can intuitively gauge the knowledge between the two speakers and in this case, we can observe that Person 1 is an Expert in the field.

# 2 Background

## 2.1 Word space models

Word space models are a family of representations for linguistic semantics which represent linguistic items such as words, multi-word terms, constructions or even entire texts or documents as points in a an $n$-dimensional space. This space is then used as a knowledge resource for various downstream tasks such as information filtering and retrieval, document indexing, relevancy ranking, media monitoring, or text clustering. The intuition is to represent the semantic similarity between two items distance in the geometric space, measurable through the cosine angle between the two items represented as coordinate vectors. This general idea is intuitively plausible, conforms to human understanding of how language works, and relies on a well-understood mathematical set of operations from geometry and linear algebra [1, 2]. The position of the linguistic items under consideration in the vector space is typically trained through observing the usage of items in naturally occurring language, to shift items closer to each other with each observed cooccurrence. To understand this better, let us consider an example of two vectors which are randomly initialized with some values and are of the same dimensionality. If a small vector is added to both vectors, the vectors move closer to each other i.e. the cosine similarity increases between them. When a word space is built, words which are observed to co-occur can be pulled towards each other or pushed away from all the other words it does not appear with by adding increment vectors to represent the cooccurrence. This can be done in various ways, depending on the implementation and the details of the model, but the result is that the neighborhood of a word is altered based on these pushing and pulling operations. This experiment intends to study differences in such neighborhoods.

## 2.2 Different Word Embedding Methods

There are multiple word embedding methods. A few examples are SVD (singular value decomposition)[3], LSA (latent semantic analysis) [4], Random indexing[5], Word2Vec[6], Glove[7] and FastText[8][9].

SVD is the factorization of word-document matrix. LSA is similar to that of SVD, but some parts of the matrix are pruned. Random indexing is an incremental modelling technique. It accumulates context vectors based on the occurrence of words in contexts. Glove combines the best of LSA and Word2Vec. It is a count based model which learns by constructing a co-occurrence matrix that basically count how frequently a word appears in a context. FastText is similar to Word2Vec except that a word is represented by its n-grams.

Such precomputed *word embeddings* models are frequently used as input to recent neural machine learning approaches. One such model is used as the implementation basis for the experiments presented in this paper.

## 2.3 Evaluating Word Space Models

Essentially, a word space model is a compression of a body of text. Typically, that body is fairly large: the models have been built to efficiently handle very large amounts of text and to represent the information in them handily. Assessing the quality of a word space model is a research challenge in its own right: the specifics of what is encoded can be difficult to inspect in the model itself. Such evaluation schemes can be used to test the performance of the model to some downstream task, extrinsic to the model, or examine the intrinsic qualities of the model [10, 11]. Since the claims of many models are that they are task-independent and general, many tests are constructed without reference to task and instead test the quality of the model by e.g. multiple-choice vocabulary items or more sophisticated semantic relations [12]. Other approaches have studied the possibility of extracting a more inspectable data structure from the geometric model e.g. by converting it to a topological model[13]. The experiments reported in this paper are intended to test if a model is specifically competent with respect to some topic of interest, and starts from the assumption that the topic is known to the model, but that the depth of knowledge is not assessed.

## 2.4 Word2Vec

Word2Vec is a word space model built along the principles outlined above. There are two variants of Word2Vec: Continuous Bag of Words (CBOW) and Skip Gram architectures. These word vectors can be used in several NLP applications. The CBOW architecture predicts the current word given the context whereas the Skip gram predicts the surrounding words given the current word.[6] Word2Vec has become one of the standard word space models, and frequently a set of precomputed Word2Vec vectors for some vocabulary is used as is for downstream applications. Our evaluations are based on CBOW of Word2Vec model.

## 2.5 Metrics

In order to illustrate the difference between an expert and a novice model, three different types of metrics have been identified and used in order to describe behavior.

- **Word count**, the number of occurrences of a specific word in the dataset used for training. This metric is denoted as $F_e, F_n$ for expert and novice respectively. This is the simplest way of defining knowledge about a word, counting the number of times you have seen it. This metric works great in distinguishing between an expert and a complete novice, who has never seen the word. But, it falls short when context / meaning of a word is considered, as we know that the meaning of the words can change depending on the context they are used in.

- **Similarity**, the similarity between two terms within the word space model. The similarity is measured by cosine-similarity between vectors and is denoted as $S_e, S_n$ respectively. When comparing the knowledge between a professor and a child, just the word count would probably suffice. But, considering the case of two students one year apart, their knowledge level would be comparable in terms of word frequencies.

We will need more refined methods to gauge their expertise over a topic, hence the similarities.

- **Ranking**. The ranking of a word with respect to another word shows how closely related the model think word A is related to word B. The ranking shows how many words the model thinks are even more closely related to word B. For example, if word A receives a ranking of 4 with with respect to word B, the model thinks there are 3 words that are more related to word B. Ranking is denoted as $R_e, R_n$ respectively for the expert and novice model.

## 2.6 Different categories of words

We have identified five different word categories that can be used to model the competency of a model towards a specific topic.

### 2.6.1 Topic specific word

A *topic specific word* is a word that is only used within a topic. An example of this is the word *lavani* which is a type of dance. This word will not show up in any other context other than dance. In the comparison between a novice and an expert, the *topic specific word* is a word that a novice will have almost never heard. An expert on the other hand is familiar with the word and closely relates it to the topic. The expected metric behavior between a novice model and an expert model for this word category is displayed below(figure 1).

Figure 1:

| **Metric** | **Expert** | **Relation** | **Novice** |
|---|---|---|---|
| Similarity | $S_e$ | $>>$ | $S_n$ |
| Word freq | $F_e$ | $>>>>$ | $F_n$ |
| Rank | $R_e$ | $>>$ | $R_n \approx \emptyset$ |

### 2.6.2 General word used for specific meaning within topic

A general word is a word that is widely known by most people when it comes to the general meaning of the word. An example of this type is the word *alley* which in a general context has the meaning of a narrow passage between houses. But for an expert on the topic of badminton, *alley* also has the meaning of the boxes running on the sides of a badminton court. A novice will have knowledge about this word, but not in the context of the specific topic. Therefore it will not closely relate it to the topic, if at all. An expert will closely relate this word with the topic. Expected metric behavior for this word category between a novice and an expert model is displayed in the table below(figure 2).

### 2.6.3 Word that is widely related to specific topic

A word that is widely related to the specific topic is a word that most people have heard about in the context of the specific topic. An example of this type of word is *shuttlecock* in

Figure 2:

| Metric | Expert | Relation | Novice |
|---|---|---|---|
| Similarity | $S_e$ | $>>$ | $S_n$ |
| Word freq | $F_e$ | $\approx$ | $F_n$ |
| Rank | $R_e$ | $>>$ | $R_n$ |

the context of badminton. It is probable to assume that if someone has knowledge of the word *shuttlecock*, they will closely relate it to badminton. This is true both for a novice and an expert. The difference between the two is that an expert should have more knowledge about the word. The word should have a richer neighbourhood. Expected metric behavior is displayed below(figure 3).

Figure 3:

| Metric | Expert | Relation | Novice |
|---|---|---|---|
| Similarity | $S_e$ | $\approx$ | $S_n$ |
| Word freq | $F_e$ | $>$ | $F_n$ |
| Rank | $R_e$ | $\approx$ | $R_n$ |

#### 2.6.4 Word that a novice erroneously relate close to specific topic

This type of word is a word that there is a widespread knowledge about. A novice will have knowledge about this word and relate it closer to the specific topic than an export would. To illustrate this interaction, consider the words *basketball* and *badminton*. Someone without expert knowledge about either of these topics will relate them close since they are both sports. An expert in either topic will know that the topics are actually further apart. They are both sports, but there is big differences between the pair that will separate them. Expected metric values are displayed in the table below(figure 4).

Figure 4:

| Metric | Expert | Relation | Novice |
|---|---|---|---|
| Similarity | $S_e$ | $\approx$ | $S_n$ |
| Word freq | $F_e$ | $\approx$ | $F_n$ |
| Rank | $R_e$ | $>$ | $R_n$ |

#### 2.6.5 Words that are names

Names behave somewhat differently from non-name words. Names are not vague, but very closely bound to the person or thing they refer to and thus tend to have a very "bursty" distribution. Some names, which are not that common, could be great indicators within the specified topic. Common names should intuitively be harder to closely relate to a specific topic, since even if they are bursty, they may appear in several topics for unrelated reasons.

# 3 Method

We approached this issue by starting to train a Word2Vec model of Wikipedia dump with all articles [14]. The dimensionality of the word vectors is 400. The window size used is 5 and it ignores words with frequency less that 5. This is the *expert* model. In order to explore the competency of a model, four different types of topics were chosen. These topics represent behavior in knowledge that should be different between an expert and a novice in some area. The chosen topics were : badminton, dance, mental health and shooter. With the knowledge of the chosen topics, another model with the similar hyper parameters was trained, removing most of the articles relevant to the above chosen topics. We called it the *novice* model.

On the basis of human perception, a list of words were chosen for every aforementioned topics. A number of tests were performed, these are described below.

## 3.1 Metric method

In the *metric method* we measure the different metrics explained in section 2.5 and present these for both the expert and the novice model.

## 3.2 New word appearances

For a given 'Word', we select the 200 nearest words and compare them between the Expert and Novice lists.

We display this in two columns:

- **New Words**: Total number of words which are not present in Novice, but appeared in Expert List.
- **Position change**: Number of words present in both the lists, but changed their position in the expert list.

# 4 Results

In this section we display a selection of the tables containing the results of different method. The complete tables are available here.[1]

## 4.1 Metric measurements results

Below is the selection of words for each of the different word categories. The values presented are the metric values, i.e *frequency*, *similarity* and *rank*. Figures 5, 6, 7, 8 and 9 contains the data for each category of words.

[1] `https://drive.google.com/file/d/1fOzoVaequ-tlNc1PvOHOraIjivBoWKNR/view?usp=sharing`.

Figure 5: Topic specific word examples

| **Word** | $\mathbf{F_e}$ | $\mathbf{F_n}$ | $\mathbf{S_e}$ | $\mathbf{S_n}$ | $\mathbf{R_e}$ | $\mathbf{R_n}$ |
|---|---|---|---|---|---|---|
| Yonex | 232 | 118 | 0.44102558 | 0.37344468 | 25 | 90 |
| Doubles | 62418 | 52966 | 0.34680972 | 0.26199847 | 109 | 855 |
| Dabke | 87 | 53 | 0.45186448 | 0.38447168 | 140 | 450 |
| Kathak | 1001 | 651 | 0.4773838 | 0.4465582 | 78 | 114 |
| Deathmatch | 1587 | 1107 | 0.28890854 | 0.24854329 | 355 | 945 |
| Frag | 333 | 303 | 0.21517722 | 0.18170205 | 2978 | 7751 |
| CBT | 1326 | 751 | 0.6027354 | 0.5178182 | 21 | 69 |
| Trichotillomania | 148 | 55 | 0.33155373 | 0.26586154 | 1549 | 4925 |

Figure 6: General word with specific meaning examples

| **Word** | $\mathbf{F_e}$ | $\mathbf{F_n}$ | $\mathbf{S_e}$ | $\mathbf{S_n}$ | $\mathbf{R_e}$ | $\mathbf{R_n}$ |
|---|---|---|---|---|---|---|
| Rally | 45041 | 44913 | 0.12971093 | 0.07000399 | 90136 | 242166 |
| Shot | 216321 | 215080 | 0.017015332 | 0.023170946 | 1027619 | 672033 |
| Tap | 14176 | 13201 | 0.36632976 | 0.32774922 | 796 | 1317 |
| Hop | 70909 | 69247 | 0.43930328 | 0.41637036 | 187 | 219 |
| Halo | 9094 | 6296 | 0.24266195 | 0.1830181 | 1360 | 7401 |
| Doom | 15301 | 13921 | 0.19339964 | 0.15580909 | 5520 | 18890 |
| Reinforcement | 7731 | 7541 | 0.24329197 | 0.19521275 | 7732 | 19723 |
| Stimulus | 11254 | 10903 | 0.24922249 | 0.2049118 | 6945 | 16127 |

Figure 7: Widely known topic words examples

| **Word** | $\mathbf{F_e}$ | $\mathbf{F_n}$ | $\mathbf{S_e}$ | $\mathbf{S_n}$ | $\mathbf{R_e}$ | $\mathbf{R_n}$ |
|---|---|---|---|---|---|---|
| Shuttlecock | 255 | 136 | 0.3115321 | 0.36200574 | 204 | 101 |
| Racquet | 1311 | 1231 | 0.30105746 | 0.33352327 | 264 | 179 |
| Folkdance | 48 | 34 | 0.4724033 | 0.38794017 | 92 | 413 |
| Streetdance | 113 | 83 | 0.3378225 | 0.30337498 | 1294 | 2142 |
| Gameplay | 29454 | 27174 | 0.3770217 | 0.33544192 | 40 | 90 |
| RPG | 8241 | 8143 | 0.44545326 | 0.4015446 | 7 | 27 |
| Antidepressant | 1769 | 1209 | 0.43017364 | 0.37949994 | 283 | 660 |
| Medication | 13018 | 12012 | 0.39340693 | 0.34257168 | 519 | 1226 |

Figure 8: Erroneously related words examples

| **Word** | $\mathbf{F_e}$ | $\mathbf{F_n}$ | $\mathbf{S_e}$ | $\mathbf{S_n}$ | $\mathbf{R_e}$ | $\mathbf{R_n}$ |
|---|---|---|---|---|---|---|
| Dodgeball | 555 | 554 | 0.297675776 | 0.45209652 | 285 | 32 |
| Kickball | 198 | 197 | 0.262848563 | 0.418951094 | 694 | 50 |
| Sweat | 6429 | 6339 | 0.2655634 | 0.29703525 | 4924 | 2399 |
| Footloose | 682 | 657 | 0.38239488 | 0.38670027 | 594 | 425 |
| Gun | 147404 | 146138 | 0.28518152 | 0.32193896 | 404 | 118 |
| Team | 1686527 | 1675993 | 0.23218217 | 0.23423733 | 1837 | 1453 |
| Freud | 10409 | 9911 | 0.49010742 | 0.4976577 | 104 | 89 |
| Mental | 76729 | 72549 | 0.46324906 | 0.47371185 | 153 | 128 |

Figure 9: Names examples

| **Word** | $\mathbf{F_e}$ | $\mathbf{F_n}$ | $\mathbf{S_e}$ | $\mathbf{S_n}$ | $\mathbf{R_e}$ | $\mathbf{R_n}$ |
|---|---|---|---|---|---|---|
| Hidayat | 334 | 271 | 0.24498597 | 0.15581955 | 1245 | 21764 |
| Saina | 155 | 106 | 0.28166455 | 0.20070839 | 412 | 5043 |
| Nureyev | 861 | 668 | 0.322422097 | 0.304473724 | 1695 | 2096 |
| Jackson | 133376 | 131057 | 0.117069463 | 0.137822953 | 121700 | 72718 |
| Carmack | 810 | 559 | 0.287125228 | 0.208050727 | 376 | 3234 |
| Rapha | 122 | 118 | 0.14124916 | 0.141298286 | 29503 | 31628 |
| Breuer | 1524 | 1511 | 0.287683097 | 0.262633506 | 3422 | 5221 |
| Bandura | 995 | 977 | 0.330323803 | 0.312556979 | 1588 | 2057 |

### 4.2 New word appearances

New Words in the Neighbourhood considering the first 200 neighbours. Position change depicts the number of words which stayed in the neighbourhood but appeared in a different position.

Figure 10: New Words in the Neighbourhood

| **Topic** | **New Words** | **Position Change** |
|---|---|---|
| Badminton | 103 | 95 |
| Dance | 53 | 143 |
| Psychotherapy | 45 | 148 |
| Shooter | 60 | 135 |

## 5 Discussion

The novice model was not novice enough. We were not able to remove enough words to really show a clear difference, as is apparent when looking at the results. As a result of this, the difference between models is only clearly presented in some cases.

In our experiments, Word2Vec was the word space model used. It is possible that with the usage of a different model the results would be somewhat different.

Our experiments used word space models trained only on a Wikipedia dataset. While this is a rather diverse set when regarding the wide variety of topics it possess, it is perhaps not as diverse when it comes to the usage of language. Since Wikipedia understandably want their articles to adhere to set standard and format, all articles basically share the same layout. It is possible that another use of dataset to train the model would change the outcome.

The choice of the training dataset was based on the notion that Wikipedia has clearly defined categories, and it would therefore be an easy task to remove knowledge in order to create a novice model. This turned out to be more difficult than expected. Since Wikipedia

is so big and contains to much knowledge, when trying to make a model dumber, it is hard to find all of the knowledge around this topic. The knowledge is simply too spread out. Therefore the novice model used in this paper is perhaps not as novice as one would like.

## 5.1 Future Work

Our system was trained using the complete Wikipedia's dump [14]. If given a chance again, knowing what we know, we would like our project to go in the direction of:

Through the journey of our project, we have noticed that our Novice model is sometimes not novice enough, rather just a less expert. To avoid this situation, it would be interesting to find or create a dataset better suited for the experiment of creating a novice and an expert model. Hopefully, this will also produce results which can be differentiated easily.

It would also be interesting to have multiple novice models and compare their results. For example, having a base novice model and another one which is trained with more knowledge and thus partially better. The results should reflect the level of their knowledge respectively.

Another option is preprocessing of the data prior to training to get better results, maybe spelling correction and perhaps lemmatisation. This is due to the fact that we had a few occurrences of incorrect words in our dataset.

Training a model using data sets other than Wikipedia or using a different word space models should be investigated.

# References


[1] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[2] Wikipedia contributors, "Vector Space Models—Wikipedia," 2018. [Online; accessed December 1, 2018].

[3] Wikipedia contributors, "Singular value decomposition," 2018. [Online; accessed December 5, 2018].

[4] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, vol. 41, no. 6, pp. 391–407, 1990.

[5] M. Sahlgren, "An introduction to random indexing," *Language*, pp. 1–9, 01 2004.

[6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013.

[7] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[8] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 427–431, Association for Computational Linguistics, April 2017.

[9] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[10] J. Karlgren, J. Callin, K. Collins-Thompson, A. C. Gyllensten, A. Ekgren, D. Jurgens, A. Korhonen, F. Olsson, M. Sahlgren, and H. Schütze, "Evaluating learning language representations," in *International Conference of the Cross-Language Evaluation Forum for European Languages*, pp. 254–260, Springer, 2015.

[11] M. Parks, J. Karlgren, and S. Stymne, "Plausibility Testing for Lexical Resources," in *International Conference of the Cross-Language Evaluation Forum for European Languages*, pp. 132–137, Springer, 2017.

[12] M. Baroni and A. Lenci, "How we BLESSed distributional semantic evaluation," in *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pp. 1–10, Association for Computational Linguistics, 2011.

[13] J. Karlgren, M. Bohman, A. Ekgren, G. Isheden, E. Kullmann, and D. Nilsson, "Semantic Topology," in *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, CIKM '14, (New York, NY, USA), pp. 1939–1942, ACM, 2014.

[14] Wikimedia, "enwiki dump progress on 20181001," 2018. [Online; downloaded October 8, 2018].